

**SVEUČILIŠTE U ZAGREBU**  
**PRIRODOSLOVNO–MATEMATIČKI FAKULTET**  
**MATEMATIČKI ODSJEK**

Ivan Kulušić

**DINAMIČKA MODALNA**  
**DEKOMPOZICIJA U ANALIZI**  
**FINANCIJSKIH TRANSAKCIJA I**  
**ALGORITAMSKOM TRGOVANJU**

Diplomski rad

Voditelj rada:  
Prof. dr. sc. Zlatko Drmač

Zagreb, veljača, 2019.

Ovaj diplomski rad obranjen je dana \_\_\_\_\_ pred ispitnim povjerenstvom u sastavu:

1. \_\_\_\_\_, predsjednik
2. \_\_\_\_\_, član
3. \_\_\_\_\_, član

Povjerenstvo je rad ocijenilo ocjenom \_\_\_\_\_.

Potpisi članova povjerenstva:

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_

# Sadržaj

<b>Sadržaj</b>	<b>iii</b>
<b>Uvod</b>	<b>1</b>
<b>1 SVD, POD i DMD</b>	<b>2</b>
1.1 SVD (Singular Value Decomposition) . . . . .	2
1.2 Generalizirani inverz . . . . .	6
1.3 POD (Proper Orthogonal Decomposition) . . . . .	9
1.4 DMD (Dynamic Mode Decomposition) . . . . .	19
<b>2 Primjena DMD-a u algoritamskom trgovanju</b>	<b>30</b>
2.1 Algoritamsko trgovanje - općenito . . . . .	30
2.2 Priprema podataka za DMD . . . . .	31
2.3 Trening algoritma za trgovanje . . . . .	32
2.4 Mogući algoritmi za trgovanje . . . . .	33
<b>3 Proširena dinamička modalna dekompozicija</b>	<b>36</b>
3.1 Koopmanov operator . . . . .	36
3.2 EDMD (Extended Dynamic Mode Decomposition) . . . . .	39
3.3 Veza između EDMD-a i DMD-a . . . . .	42
<b>Bibliografija</b>	<b>44</b>

# Uvod

Algoritamsko trgovanje sve više dobiva na važnosti pri odabiru sofisticiranih investicijskih strategija. Otprilike polovicu svih transakcija na svjetskom tržištu dionica provode automatizirani programi za trgovanje. Velik dio tih transakcija otpada na visokofrekventno trgovanje (kupnja ili prodaja svake sekunde ili milisekunde). Algoritamsko trgovanje pokreću matematički modeli koji pokušavaju shvatiti uzorak kretanja cijena na tržištu. Ukoliko matematički model točno predviđa kuda će cijene ići, tu informaciju možemo iskoristiti pri odluci o kupnji ili prodaji. Većina sofisticiranih algoritama se oslanja na statističke i vjerojatnosne modele pomoću kojih se određuju mjere rizika i pouzdani intervali te nam te informacije pomažu pri trgovanju (primjerice VaR - Value at Risk).

U ovom radu je opisan algoritam za automatsko trgovanje baziran na pretpostavci da su cijene na tržištu pokretane nelinearnim dinamičkim sustavom. Na financijske podatke s tržišta dionica (cijene dionica) primjenjujemo dinamičku modalnu dekompoziciju (DMD). DMD metoda je matična dekompozicija temeljena na SVD-u (eng. Singular Value Decomposition) koja izvlači vremensko-prostornu strukturu dinamičkog sustava. Ova metoda ne zahtijeva jednadžbu sustava, već je u potpunosti pokretana podacima. Metoda uspješno rekonstruira dinamiku direktno iz podataka generiranih sustavom. Nelinearni sustav lokalno aproksimira linearnim sustavom nižeg reda kojeg možemo iskoristiti za aproksimaciju stanja nelinearnog sustava u slijedećih nekoliko trenutaka. Takva predviđanja će nam pomoći pri izradi strategije za trgovanje. DMD parametriziramo da bi dobili algoritam strojnog učenja koji se je sposoban stalno prilagođavati novoj dinamici tržišta kako vrijeme prolazi. Naš algoritam strojnog učenja određuje optimalan broj podataka koje metoda uzima u obzir za izračun cijena slijedećih nekoliko dana, također ćemo odrediti koji je najoptimalniji broj dana u budućnosti za koje ćemo aproksimirati cijene.

Prije nego prijedemo na opis algoritma za trgovanje, dat ćemo detaljan uvod u DMD metodu i sam algoritam računanja DMD modova u Matlabu. Također ćemo opisati i POD (eng. Proper Orthogonal Decomposition) metodu koja nam pomaže pri aproksimaciji sustava sustavom nižeg reda te ćemo prikazati neka od svojstava SVD dekompozicije i generaliziranog inverza koji će nam poslužiti kao tehnički alat za izračun POD-a i DMD-a.

# Poglavlje 1

## SVD, POD i DMD

### 1.1 SVD (Singular Value Decomposition)

SVD je jedna od najvažnijih matricnih dekompozicija u numeričkoj analizi. Kako s teoretske strane, za dokazivanje činjenica, tako i s praktične strane, za razvoj numeričkih algoritama. Metode POD (eng. Proper Orthogonal Decomposition) i DMD (eng. Dynamic Mode Decomposition) se temelje na SVD dekompoziciji. Slijedeći teorem garantira postojanje SVD-a.

**Teorem 1.1.1.** *Neka je  $M \in \mathbb{C}^{m \times n}$  kompleksna matrica,  $m \geq n$ .  $M$  se može rastaviti kao*

$$M = U \Sigma V^*, \text{ gdje su } U \in \mathbb{C}^{m \times m}, V \in \mathbb{C}^{n \times n} \text{ unitarne}^1 \text{ matrice, } \Sigma = \begin{bmatrix} \Sigma_1 \\ 0 \end{bmatrix}$$

$$\Sigma_1 = \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_n \end{bmatrix}, \text{ takva da } \sigma_1, \dots, \sigma_n \in \mathbb{R}, \quad \sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n \geq 0.$$

*Vrijednosti  $\sigma_1, \dots, \sigma_n$  nazivamo singularnim vrijednostima matrice  $M$ .*

*Ukoliko je  $M$  realna matrica, onda su matrice  $U$  i  $V$  realne ortogonalne<sup>2</sup>. Ukoliko je  $m < n$ , računamo SVD matrice  $M^*$  te hermitskim adjungiranjem SVD-a matrice  $M^*$  dobivamo SVD matrice  $M$ .*

*Dokaz.* Dokaz teorema se može pronaći u [2].

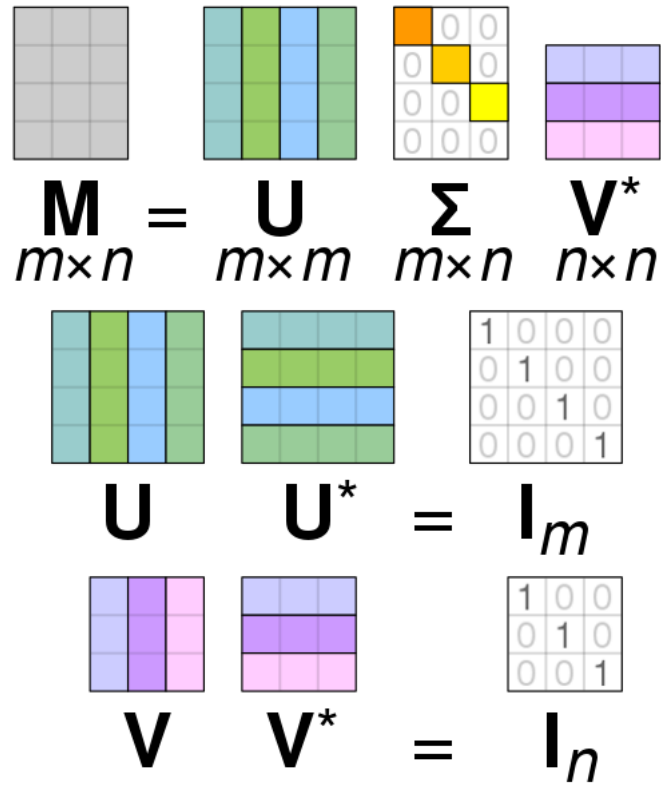
□

---

<sup>1</sup>Za matricu  $A \in \mathbb{C}^{k \times k}$  kažemo da je unitarna ukoliko vrijedi  $AA^* = A^*A = I$ , gdje je  $A^*$  hermitski adjungirana matrica matrice  $A$  i vrijedi  $A_{ij}^* = \bar{A}_{ji}$  (predstavlja kompleksno konjugiranje broja).

<sup>2</sup>Za matricu  $A \in \mathbb{R}^{k \times k}$  kažemo da je ortogonalna ukoliko vrijedi  $AA^T = A^T A = I$ , gdje je  $A^T$  transponirana matrica matrice  $A$  i vrijedi  $A_{ij}^T = A_{ji}$ .

Slijedeća slika ilustrira SVD.



Slika 1.1: Shematski prikaz SVD-a<sup>3</sup>.

<sup>3</sup>Slika preuzeta s [https://en.wikipedia.org/wiki/Singular\\_value\\_decomposition#/media/File:Singular\\_value\\_decomposition\\_visualisation.svg](https://en.wikipedia.org/wiki/Singular_value_decomposition#/media/File:Singular_value_decomposition_visualisation.svg).

Fundamentalni pojmovi<sup>4</sup> iz linearne algebre koji će se provlačiti kroz rad:

1. Euklidski (standardni) skalarni produkt

Neka su  $x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$  i  $y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$  vektori iz  $\mathbb{R}^n$ , njihov standardni skalarni produkt definiramo kao<sup>5</sup>

$$\langle x, y \rangle_{\mathbb{R}^n} := x_1 y_1 + x_2 y_2 + \dots + x_n y_n = \sum_{i=1}^n x_i y_i = x^T y.$$

2. Euklidska (standardna) norma

Standardnu normu vektora  $x \in \mathbb{R}^n$  definiramo kao<sup>6</sup>

$$\|x\|_{\mathbb{R}^n} := \sqrt{\langle x, x \rangle_{\mathbb{R}^n}} = \sqrt{x^T x} = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2}.$$

3. Ortonormirana baza

Za bazu  $\{e_1, e_2, \dots, e_n\}$  vektorskog prostora  $\mathbb{R}^n$  (ili  $\mathbb{C}^n$ ) kažemo da je ortonormirana ukoliko vrijedi

$$\langle e_i, e_j \rangle_{\mathbb{R}^n(\mathbb{C}^n)} = \begin{cases} 1, & \text{ako je } i = j \\ 0, & \text{ako je } i \neq j \end{cases}, \quad i, j \in \{1, 2, \dots, n\}.$$

4. Ortogonalni komplement

Neka je  $V$  vektorski prostor ( $\mathbb{R}^n, \mathbb{C}^n, \dots$ ) sa skalarnim produktom  $\langle \cdot, \cdot \rangle$ . Ortogonalni komplement potprostora  $M \leq V$  definiramo kao

$$M^\perp := \{x \in V : \langle x, v \rangle = 0, \forall v \in M\}.$$

5. Slika i jezgra operatora

Za danu matricu  $A \in \mathbb{R}^{m \times n}$  (analogno za  $A \in \mathbb{C}^{m \times n}$  uz zamjenu  $\cdot^T \rightarrow \cdot^*$ ) definiramo skupove

$$\mathcal{N}(A) = \{x \in \mathbb{R}^n : Ax = 0\} \text{ i } \mathcal{R}(A) = \{b \in \mathbb{R}^m : b = Ax \text{ za neki } x \in \mathbb{R}^n\}.$$

$\mathcal{R}(A)$  nazivamo slika linearnog operatora  $L : \mathbb{R}^n \rightarrow \mathbb{R}^m$  definiranog matricom  $A$ ,  $L(x) = Ax$ .  $\mathcal{N}(A)$  nazivamo jezgrom linearnog operatora  $L$ . Vrijedi:

<sup>4</sup>Pojmove kao što su vektorski prostor, potprostor vektorskog prostora, baza vektorskog prostora, linearni operator i slične, nećemo posebno definirati.

<sup>5</sup>Za vektore  $x, y \in \mathbb{C}^n$  definiramo  $\langle x, y \rangle_{\mathbb{C}^n} := \bar{x}_1 y_1 + \bar{x}_2 y_2 + \dots + \bar{x}_n y_n = \sum_{i=1}^n \bar{x}_i y_i = x^* y$ .

<sup>6</sup>Za vektor  $x \in \mathbb{C}^n$  definiramo  $\|x\|_{\mathbb{C}^n} := \sqrt{\langle x, x \rangle_{\mathbb{C}^n}} = \sqrt{x^* x} = \sqrt{\bar{x}_1 x_1 + \bar{x}_2 x_2 + \dots + \bar{x}_n x_n} = \sqrt{|x_1|^2 + |x_2|^2 + \dots + |x_n|^2}.$

- $\mathcal{N}(A)$  i  $\mathcal{R}(A^T)$  su potprostori od  $\mathbb{R}^n$ ,
- $\mathcal{R}(A)$  i  $\mathcal{N}(A^T)$  su potprostori od  $\mathbb{R}^m$ ,
- $\mathcal{N}(A) = \mathcal{R}(A^T)^\perp$ ,
- $\mathcal{N}(A^T) = \mathcal{R}(A)^\perp$ .

#### 6. Ortogonalni projektor

Neka je  $V$  potprostor prostora  $\mathbb{R}^n$  (analogno za  $\mathbb{C}^n$ ). Vektor  $x \in \mathbb{R}^n$  možemo na jedinstven način prikazati kao  $x = v + u$ , gdje je  $v \in V$  i  $u \in V^\perp$ .<sup>7</sup> Linearni operator  $P_V$  definiran s  $P_V x := v$  nazivamo ortogonalni projektor na potprostor  $V$ . Vrijedi:

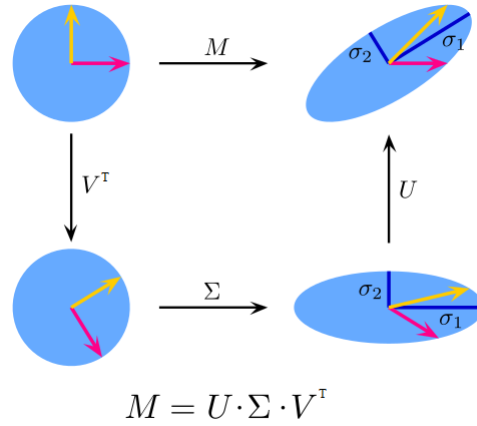
- $\|x - P_V x\|_{\mathbb{R}^n} = \|x - v\|_{\mathbb{R}^n} = \min_{y \in V} \|x - y\|_{\mathbb{R}^n}$ ,
- Neka je  $A \in \mathbb{C}^{n \times k}$  kompleksna matrica čiji stupci čine bazu  $k$ -dimenzionalnog potprostora od  $V$ . Ortogonalni projektor na potprostor razapet stupcima od  $A$  je dan s  $P_A = A(A^*A)^{-1}A^*$ . Ako su stupci matrice  $A$  ortonormirani, onda je ortogonalni projektor na isti potprostor dan s  $P_A = AA^*$ .

### Intuitivna interpretacija dekompozicije singularnih vrijednosti

- Ako o matrici  $M \in \mathbb{R}^{m \times n}$  razmišljamo kao o zapisu operatora koji preslikava vektor  $x \in \mathbb{R}^n$  u vektor  $y = Mx \in \mathbb{R}^m$ , onda možemo odabrati ortonormiranu bazu u  $\mathbb{R}^n$  (stupci matrice  $V$ ) i ortonormiranu bazu u  $\mathbb{R}^m$  (stupci matrice  $U$ ) takve da je zapis operatora  $M$  u paru tih baza “dijagonalna” matrica  $\Sigma$ . Ovdje o matrici  $V^T$  razmišljamo kao o matrici prijelaza iz baze na  $\mathbb{R}^n$  u kojoj je zapisan linearni operator (najčešće kanonska baza) u bazu koju čine stupci matrice  $V$ . O matrici  $U$  razmišljamo kao o matrici prijelaza iz baze koju čine stupci matrice  $U$  u bazu na  $\mathbb{R}^m$  u kojoj je originalno zapisan operator  $M$ .
- Na slijedećoj slici vidimo da djelovanje bilo kojeg linearnog operatora  $M$  na vektor  $x \in \mathbb{R}^n$  možemo pomoću SVD-a rastaviti na tri koraka, odnosno kompoziciju tri jednostavnija linearna preslikavanja:
  1. Rotacija: preslikavanje  $V^T$  rotira vektor  $x$ .
  2. Razvlačenje prostora: preslikavanje  $\Sigma$  djeluje na zarotirani vektor.
  3. Rotacija: preslikavanjem  $U$  zakrenemo dobiveni vektor.

<sup>7</sup>Ovo je tvrdnja teorema koji se dokazuje na uvodnim kolegijima u linearnu algebru.



Slika 1.2: Djelovanje operatora  $M^8$ .

- POD će nam dati još jednu intuitivnu interpretaciju matrice  $U$  (Sekcija 1.3).

## 1.2 Generalizirani inverz

Kao što mu i samo ime govori, generalizirani inverz ili Moore-Penroseov inverz je generalizacija inverza matrice.

### Definicija 1.2.1. Generalizirani inverz.

Neka je  $M \in \mathbb{C}^{m \times n}$  kompleksna matrica. Njezin generalizirani inverz je jedinstvena matrica  $M^\dagger \in \mathbb{C}^{n \times m}$  takva da zadovoljava slijedeće četiri jednakosti:

$$MM^\dagger M = M, \quad (1.1)$$

$$M^\dagger MM^\dagger = M^\dagger, \quad (1.2)$$

$$(MM^\dagger)^* = MM^\dagger, \quad (1.3)$$

$$(M^\dagger M)^* = M^\dagger M. \quad (1.4)$$

Uvjete (1.1), (1.2), (1.3) i (1.4) zovemo **Moore-Penroseovi kriteriji**.

<sup>8</sup>Slika preuzeta s [https://en.wikipedia.org/wiki/Singular\\_value\\_decomposition#/media/File:Singular-Value-Decomposition.svg](https://en.wikipedia.org/wiki/Singular_value_decomposition#/media/File:Singular-Value-Decomposition.svg).

**Napomena 1.2.1.** (Alternativne definicije.)1. Funkcijska definicija

Neka je  $M \in \mathbb{C}^{m \times n}$  kompleksna matrica i neka je  $\tilde{M} : \mathbb{C}^n \rightarrow \mathbb{C}^m$  linearna transformacija definirana matricom  $M$ ,  $\tilde{M}(x) = Mx$ ,  $x \in \mathbb{C}^n$ . Neka je linearna transformacija  $\tilde{M}^\dagger : \mathbb{C}^m \rightarrow \mathbb{C}^n$  dana s

$$\tilde{M}^\dagger(y) = \begin{cases} 0, & \text{ako je } y \in \mathcal{R}(M)^\perp \\ (\tilde{M}|_{\mathcal{R}(M^*)})^{-1}y, & \text{ako je } y \in \mathcal{R}(M).^9 \end{cases}$$

Matrica linearne transformacije  $\tilde{M}^\dagger$ , u oznaci  $M^\dagger$ , zove se **generalizirani inverz** matrice  $M$ .

2. Mooreova definicija generaliziranog inverza

Neka je  $M \in \mathbb{C}^{m \times n}$  kompleksna matrica. Njezin **generalizirani inverz** je jedinstvena matrica  $M^\dagger \in \mathbb{C}^{n \times m}$  takva da je

$$MM^\dagger = P_{\mathcal{R}(M)} \quad \text{i} \quad M^\dagger M = P_{\mathcal{R}(M^*)}.$$

$P_X$  označava ortogonalni projektor na potprostor  $X$  prostora  $\mathbb{C}^m$ , odnosno  $\mathbb{C}^n$ .

**Teorem 1.2.1.** Generalizirani inverz matrice  $M$  je jedinstven.

*Dokaz.* Pretpostavimo da postoje dva različita generalizirana inverza  $B$  i  $C$  koji zadovoljavaju Moore-Penroseove kriterije. Tada vrijedi:

$$\begin{aligned} MB &\stackrel{(1.1)}{=} MCMB \stackrel{(1.3)}{=} (MC)^*(MB)^* = C^*M^*B^*M^* = C^*(MBM)^* \stackrel{(1.1)}{=} C^*M^* = (MC)^* \stackrel{(1.3)}{=} MC, \\ BM &\stackrel{(1.1)}{=} BMC M \stackrel{(1.4)}{=} (BM)^*(CM)^* = M^*B^*M^*C^* = (MBM)^*C^* \stackrel{(1.1)}{=} M^*C^* = (CM)^* \stackrel{(1.4)}{=} CM. \end{aligned}$$

Iz  $MB = MC$  i  $BM = CM$  slijedi  $B \stackrel{(1.2)}{=} BMB = BMC = CMC \stackrel{(1.2)}{=} C$ , što je u kontradikciji s pretpostavkom  $B \neq C$ .

□

---

<sup>9</sup> $\mathcal{R}(M^*) = \mathcal{N}(M)^\perp \implies \tilde{M}|_{\mathcal{R}(M^*)}$  je surjekcija. Transformacija  $\tilde{M}|_{\mathcal{R}(M^*)}$  slika iz  $\mathcal{R}(M^*)$  u  $\mathcal{R}(M)$  i surjektivna je. Budući da su potprostori  $\mathcal{R}(M^*)$  i  $\mathcal{R}(M)$  iste dimenzije  $\tilde{M}|_{\mathcal{R}(M^*)}$  je i injekcija te je stoga bijekcija pa ima inverz. Inverz linearne transformacije je također linearan.

## Računanje generaliziranog inverza pomoću SVD-a

**Teorem 1.2.2.** *Postoji generalizirani inverz matrice  $M \in \mathbb{C}^{m \times n}$ .*

*Dokaz.* Generalizirani inverz matrice oblika  $\Sigma = \begin{bmatrix} \Sigma_1 \\ 0 \end{bmatrix} \in \mathbb{R}^{m \times n}$ ,  $m \geq n$ , gdje je

$$\Sigma_1 = \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_n \end{bmatrix}, \quad \sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n \geq 0$$

je dan s  $\Sigma^\dagger = \begin{bmatrix} \Sigma_1^\dagger & 0 \end{bmatrix} \in \mathbb{R}^{n \times m}$ , gdje je

$$\Sigma_1^\dagger = \begin{bmatrix} \frac{1}{\sigma_1} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{\sigma_n} \end{bmatrix} \quad (\text{ako je } \sigma_i = 0, \text{ definiramo } (\Sigma_1^\dagger)_{ii} := 0).$$

Provjerom Moore-Penroseovih kriterija se lako uvjerimo u istinitost gornje tvrdnje.<sup>10</sup>

Znamo da za svaku matricu  $M \in \mathbb{C}^{m \times n}$  postoji SVD, tj. postoje  $U, \Sigma$  i  $V$  zadanog oblika takve da  $M = U\Sigma V^*$ .

$$\text{Definirajmo } M^\dagger := V\Sigma^\dagger U^* \in \mathbb{C}^{n \times m}.$$

Opet, provjeravanjem Moore-Penroseovih kriterija, se lako uvjerimo da je ovako definirana  $M^\dagger$  generalizirani inverz proizvoljno uzete matrice  $M \in \mathbb{C}^{m \times n}$ .<sup>11</sup>

□

Prethodni teorem nam daje algoritam za računanje pseudoinverza pomoću SVD-a, dok nam teorem prije njega garantira jedinstvenost pseudoinverza.

Ako je  $M$  kvadratna regularna matrica, onda je  $M^\dagger = M^{-1}$ .

**Napomena 1.2.2.** (Važna primjena pseudoinverza.)

Neka je  $A \in \mathbb{C}^{m \times n}$  i neka nam je dan sustav linearnih jednadžbi  $Ax = b$ . U generalnom slučaju vektor  $x$  koji rješava sustav ne mora postojati ili ako postoji, ne mora biti jedinstven.

Pseudoinverz nam daje rješenje za problem najmanjih kvadrata u slijedećem smislu:

$\forall x \in \mathbb{C}^n$  imamo  $\|Ax - b\|_{\mathbb{C}^m} \geq \|Az - b\|_{\mathbb{C}^m}$ , gdje je  $z = A^\dagger b$ . Prethodna nejednakost prelazi u jednakost ako i samo ako  $x = A^\dagger b + (I - A^\dagger A)w$  za sve vektore  $w \in \mathbb{C}^n$  (Operator  $I - A^\dagger A$  je projektor na jezgru od  $A$ ). Rješenje s najmanjom euklidskom normom je upravo  $z = A^\dagger b$ .

<sup>10</sup>Za kompletan raspis pogledati [3].

<sup>11</sup>Ako je  $M \in \mathbb{R}^{m \times n}$ , onda je  $M^\dagger = V\Sigma^\dagger U^T \in \mathbb{R}^{n \times m}$ .

Vrijedi i generalizacija gornje tvrdnje:

Neka je  $B \in \mathbb{C}^{m \times p}$ .  $\forall X \in \mathbb{C}^{n \times p}$  imamo  $\|AX - B\|_F \geq \|AZ - B\|_F$  gdje je  $Z = A^\dagger B$ , a  $\|\cdot\|_F$  predstavlja Frobeniusovu normu<sup>12</sup>.

### 1.3 POD (Proper Orthogonal Decomposition)

Slijedeća razrada je preuzeta iz [9] i [4]. U ovom poglavlju uvodimo POD metodu u Euklidskom prostoru  $\mathbb{R}^m$ . Analogno se metoda generalizira i na  $\mathbb{C}^m$ , no za ilustraciju metode je dovoljno pogledati slučaj  $\mathbb{R}^m$ .

Cilj POD-a je za dani skup vektora  $y_1, y_2, \dots, y_n \in \mathbb{R}^m$  odrediti  $l$ -dimenzionalan potprostor od  $\mathbb{R}^m$  koji ih najbolje aproksimira u smislu najmanjih kvadrata. Vrijedi  $l \leq \min\{m, n\}$ . Drugim riječima, tražimo matricu  $Q \in \mathbb{R}^{m \times l}$  takvu da vrijedi

$$Q = \operatorname{argmin}_{M \in \mathbb{R}^{m \times l}} \sum_{i=1}^n \|y_i - MM^T y_i\|_{\mathbb{R}^m}^2, \quad \text{gdje je } M \text{ takva da } M^T M = I_l.$$

Stupci matrice  $Q$  su ortonormirana baza za traženi  $l$ -dimenzionalni potprostor.

U prvom dijelu ovog poglavlja imamo drugačiji pristup računanju POD baze od gore definiranog, no kasnije u napomeni 1.3.1 pokazujemo da su oba pristupa ekvivalentna.

#### Veza POD i SVD

Neka je  $Y = [y_1, \dots, y_n]$  realna  $m \times n$  matrica ranga  $d \leq \min\{m, n\}$  sa stupcima  $y_j \in \mathbb{R}^m, 1 \leq j \leq n$ .

SVD garantira postojanje realnih brojeva  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_d > 0$  i ortogonalnih matrica  $U \in \mathbb{R}^{m \times m}$  sa stupcima  $\{u_i\}_{i=1}^m$  i  $V \in \mathbb{R}^{n \times n}$  sa stupcima  $\{v_i\}_{i=1}^n$  takvih da

$$U^T Y V = \begin{bmatrix} D & 0 \\ 0 & 0 \end{bmatrix} =: \Sigma \in \mathbb{R}^{m \times n},$$

gdje je

$$D = \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_d \end{bmatrix} \in \mathbb{R}^{d \times d}.$$

<sup>12</sup> $\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |A_{ij}|^2} = \sqrt{\operatorname{tr}(AA^*)}$ , za  $A \in \mathbb{C}^{m \times n}$ .

<sup>13</sup>Operator  $L : \mathbb{R}^m \rightarrow \mathbb{R}^m$ , definiran s  $L(x) = MM^T x$  predstavlja ortogonalnu projekciju na potprostor razapet stupcima od matrice  $M$ .

Vektori  $\{u_i\}_{i=1}^d$  i  $\{v_i\}_{i=1}^d$  zadovoljavaju

$$Yv_i = \sigma_i u_i \quad i \quad Y^T u_i = \sigma_i v_i, \quad i = 1, \dots, d.$$

Vektori  $\{u_i\}_{i=1}^d$  i  $\{v_i\}_{i=1}^d$  su svojstveni vektori matrica  $YY^T$  i  $Y^T Y$ , respektivno, sa svojstvenim vrijednostima  $\lambda_i = \sigma_i^2 > 0, i = 1, \dots, d$ .

Vektori  $\{u_i\}_{i=d+1}^m$  i  $\{v_i\}_{i=d+1}^n$  su svojstveni vektori matrica  $YY^T$  i  $Y^T Y$ , respektivno, sa svojstvenom vrijednošću 0 (ako vrijedi  $d < m$  i/ili  $d < n$ ).

Ukoliko pogledamo rastav  $Y = U\Sigma V^T$ , uočavamo da zbog svojstava matrice  $\Sigma$  matricu  $Y$  možemo zapisati kao

$$Y = U^d D(V^d)^T,$$

gdje je matrica  $U^d \in \mathbb{R}^{m \times d}$  dobivena uzimanjem prvih  $d$  stupaca iz matrice  $U$ , a matrica  $V^d \in \mathbb{R}^{n \times d}$  uzimanjem prvih  $d$  stupaca iz matrice  $V$ .

Uvedimo matricu  $B^d := D(V^d)^T \in \mathbb{R}^{d \times n}$ . Matricu  $Y$  možemo zapisati u formi

$$Y = U^d B^d.$$

Iz gornjeg rastava vidimo da svaki vektor stupac  $y_i$  matrice  $Y$  možemo zapisati u obliku

$$y_i = B_{1,i}u_1 + \dots + B_{d,i}u_d, \quad i = 1, \dots, n. \quad (1.5)$$

Drugim riječima, stupci  $\{u_i\}_{i=1}^d$  čine ortonormiranu bazu  $d$ -dimenzionalnog potprostora u  $\mathbb{R}^m$  razapetog stupcima  $\{y_i\}_{i=1}^n, d \leq n$ .

Nadalje, budući da se radi o ortonormiranoj bazi, množeći relaciju (1.5) s  $u_j$  zdesna ( $j = 1, \dots, d$ ) vidimo da koeficijente  $B_{j,i}$  možemo dobiti kao

$$B_{j,i} = \langle y_i, u_j \rangle_{\mathbb{R}^m}, \quad j = 1, \dots, d, i = 1, \dots, n.$$

Sve zajedno imamo (uz zamjenu oznaka  $i \longleftrightarrow j$ )

$$y_j = \sum_{i=1}^d \langle y_j, u_i \rangle_{\mathbb{R}^m} u_i, \quad j = 1, \dots, n.$$

Za početak pronađimo jednodimenzionalan potprostor od  $\mathbb{R}^m$  koji aproksimira naše podatke. Kako odabrati ortonormiranu bazu tog potprostora?

Neka je vektor  $u$  duljine 1, odnosno  $\|u\|_{\mathbb{R}^m} = 1$ . Tada veličina  $|\langle y_j, u \rangle_{\mathbb{R}^m}|$  predstavlja komponentu vektora  $y_j$  u smjeru vektora  $u$ , odnosno duljinu ortogonalne projekcije vektora  $y_j$  na smjer  $u$ .

Kao dobra ideja se čini uzeti takav vektor  $u$  u smjeru kojega će vektori  $y_j, j = 1, \dots, n$  imati najduže komponente. Dakle, da bismo odredili bazu jednodimenzionalnog potprostora

koji aproksimira vektore  $y_j, j = 1, \dots, n$  potrebno je pronaći vektor  $u$  koji rješava slijedeći optimizacijski problem:

$$\max_{u \in \mathbb{R}^m} \sum_{j=1}^n |\langle y_j, u \rangle_{\mathbb{R}^m}|^2, \quad \text{uz uvjet } \|u\|_{\mathbb{R}^m} = 1. \quad (P^1)$$

Rješavanjem dolazimo do zaključka da vektor  $u_1$ , tj. prvi stupac matrice  $U$  iz SVD-a matrice  $Y$ , rješava gornji optimizacijski problem te maksimum iznosi  $\max(P^1) = \lambda_1 = \sigma_1^2$ . Za detalje dokaza ove tvrdnje pogledati [9].

Aproksimirajmo sada vektore  $y_j, j = 1, \dots, n$  dvodimenzionalnim potprostorom od  $\mathbb{R}^m$ , tj. odredimo ortonormiranu bazu tog potprostora. Nadogradit ćemo bazu jednodimenzionalnog potprostora  $\{u_1\}$  vektorom  $u$  koji rješava slijedeći optimizacijski problem:

$$\max_{u \in \mathbb{R}^m} \sum_{j=1}^n |\langle y_j, u \rangle_{\mathbb{R}^m}|^2, \quad \text{uz uvjete } \|u\|_{\mathbb{R}^m} = 1 \text{ i } \langle u, u_1 \rangle_{\mathbb{R}^m} = 0. \quad (P^2)$$

Rješavanjem dolazimo do zaključka da vektor  $u_2$ , tj. drugi stupac matrice  $U$  iz SVD-a matrice  $Y$ , rješava prethodni optimizacijski problem te vrijedi  $\max(P^2) = \lambda_2 = \sigma_2^2$ .

Po indukciji možemo nastaviti dalje. Slijedeći teorem sumira prethodne zaključke i daje direktnu vezu između POD-a i SVD-a.

**Teorem 1.3.1.** *Neka je  $Y = [y_1, \dots, y_n] \in \mathbb{R}^{m \times n}$  dana matrica ranga  $d \leq \min\{m, n\}$ . Neka je  $Y = U\Sigma V^T$  njen SVD. Tada je za svaki  $l \in \{1, \dots, d\}$  rješenje problema*

$$\max_{\tilde{u}_1, \dots, \tilde{u}_l \in \mathbb{R}^m} \sum_{i=1}^l \sum_{j=1}^n |\langle y_j, \tilde{u}_i \rangle_{\mathbb{R}^m}|^2, \quad \text{uz uvjet } \langle \tilde{u}_i, \tilde{u}_j \rangle_{\mathbb{R}^m} = \delta_{ij}, 1 \leq i, j \leq l \quad (P^l)$$

*dano singularnim vektorima  $\{u_i\}_{i=1}^l$ , tj. s prvih  $l$  stupaca matrice  $U$ . Također vrijedi*

$$\max(P^l) = \sum_{i=1}^l \sigma_i^2 = \sum_{i=1}^l \lambda_i.$$

*$U(P^l) \delta_{ij}$  predstavlja Kroneckerov simbol, tj.*

$$\delta_{ij} = \begin{cases} 1, & \text{ako je } i = j \\ 0, & \text{ako je } i \neq j \end{cases}.$$

*Dokaz.* Za dokaz pogledati [9]. □

Motivirani prethodnom razradom i prethodnim teoremom uvodimo slijedeću definiciju:

**Definicija 1.3.1. POD baza.**

Neka je  $Y = [y_1, \dots, y_n] \in \mathbb{R}^{m \times n}$  dana matrica ranga  $d \leq \min\{m, n\}$ . Neka je  $Y = U\Sigma V^T$  njen SVD. Vektor  $u_i \in \mathbb{R}^m$  predstavlja  $i$ -ti supac matrice  $U$ ,  $i = 1, \dots, m$ . Za  $l \in \{1, \dots, d\}$  skup vektora  $\{u_i\}_{i=1}^l$  se zove POD baza ranga  $l$  danih vektora  $y_1, \dots, y_n$ .

**Korolar 1.3.2. Optimalnost POD baze.**

Neka vrijede pretpostavke teorema 1.3.1. Neka je  $\hat{U}^d \in \mathbb{R}^{m \times d}$  matrica s međusobno ortonormiranim stupcima  $\hat{u}_i$  i neka je zapis stupaca matrice  $Y$  u bazi  $\{\hat{u}_i\}_{i=1}^d$  dan s

$$Y = \hat{U}^d C^d, \quad \text{gdje su } C_{ij}^d = \langle \hat{u}_i, y_j \rangle_{\mathbb{R}^m}, \quad 1 \leq i \leq d, 1 \leq j \leq n.$$

Tada za svaki  $l \in \{1, \dots, d\}$  imamo

$$\|Y - U^l B^l\|_F \leq \|Y - \hat{U}^l C^l\|_F.$$

$\|\cdot\|_F$  označava Frobeniusovu normu koja je dana s

$$\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |A_{ij}|^2} = \sqrt{\text{trag}(A^T A)}, \quad \text{za } A \in \mathbb{R}^{m \times n}.$$

Matrica  $U^l$  označava prvih  $l$  stupaca matrice  $U$ ,  $l \leq d$ .  $B^l$  označava prvih  $l$  redaka matrice  $B$ . Analogno za  $\hat{U}^l$  i  $C^l$ .

*Dokaz.*

Promotrimo matricu  $C_0^l$  koju dobijemo iz  $C \in \mathbb{R}^{d \times n}$  tako da zamijenimo zadnjih  $d-l$  redaka s nulama. Dakle matrica  $C_0^l$  izgleda ovako:

$$C_0^l = \begin{bmatrix} C_{11}^d & \dots & C_{1n}^d \\ \vdots & & \vdots \\ C_{l1}^d & \dots & C_{ln}^d \\ 0 & \dots & 0 \\ \vdots & & \vdots \\ 0 & \dots & 0 \end{bmatrix}.$$

Uzimajući u obzir pravila množenja matrica i svojstva matrice  $C_0^l$  lako se uvjerimo da vrijedi  $\hat{U}^l C^l = \hat{U}^d C_0^l$ , iz čega slijedi:

$$\|Y - \hat{U}^l C^l\|_F^2 = \|\hat{U}^d (C^d - C_0^l)\|_F^2 = \|C^d - C_0^l\|_F^2 = \sum_{i=l+1}^d \sum_{j=1}^n |C_{ij}^d|^2.$$

U drugoj jednakosti koristimo činjenicu da je  $\hat{U}^d$  ortogonalna matrica. Također vrijedi slijedeći niz jednakosti

$$YY^T u_i = Y \begin{bmatrix} y_1^T \\ y_2^T \\ \vdots \\ y_n^T \end{bmatrix} u_i = [y_1, \dots, y_n] \begin{bmatrix} \langle y_1, u_i \rangle_{\mathbb{R}^m} \\ \langle y_2, u_i \rangle_{\mathbb{R}^m} \\ \vdots \\ \langle y_n, u_i \rangle_{\mathbb{R}^m} \end{bmatrix} = \sum_{j=1}^n \langle y_j, u_i \rangle_{\mathbb{R}^m} y_j. \quad (1.6)$$

Analogno kao matricu  $C_0^l$  uvodimo i matricu  $B_0^l$ .

Vrijedi:

$$\begin{aligned} \|Y - U^l B^l\|_F^2 &= \|U^d (B^d - B_0^l)\|_F^2 = \|B^d - B_0^l\|_F^2 = \sum_{i=l+1}^d \sum_{j=1}^n |B_{ij}^d|^2 = \\ &= \sum_{i=l+1}^d \sum_{j=1}^n |\langle y_j, u_i \rangle_{\mathbb{R}^m}|^2 = \sum_{i=l+1}^d \sum_{j=1}^n \langle \langle y_j, u_i \rangle_{\mathbb{R}^m} y_j, u_i \rangle_{\mathbb{R}^m} = \\ &= \sum_{i=l+1}^d \langle \sum_{j=1}^n \langle y_j, u_i \rangle_{\mathbb{R}^m} y_j, u_i \rangle_{\mathbb{R}^m} \stackrel{(1.6)}{=} \sum_{i=l+1}^d \langle YY^T u_i, u_i \rangle_{\mathbb{R}^m} = \\ &= \sum_{i=l+1}^d \langle \sigma_i^2 u_i, u_i \rangle_{\mathbb{R}^m} = \sum_{i=l+1}^d \sigma_i^2 \langle u_i, u_i \rangle_{\mathbb{R}^m} = \sum_{i=l+1}^d \sigma_i^2. \end{aligned}$$

Analogno prethodnom izvodu dobije se

$$\|Y\|_F^2 = \|U^d B^d\|_F^2 = \|B^d\|_F^2 = \sum_{i=1}^d \sum_{j=1}^n |B_{ij}^d|^2 = \sum_{i=1}^d \sigma_i^2.$$

Također vrijedi:

$$\|Y\|_F^2 = \|\hat{U}^d C^d\|_F^2 = \|C^d\|_F^2 = \sum_{i=1}^d \sum_{j=1}^n |C_{ij}^d|^2.$$

Konačno, budući da su vektori  $u_1, \dots, u_l$  rješenje problema  $(P^l)$  dobivamo slijedeću nejednakost koja dokazuje korolar.

$$\begin{aligned} \|Y - U^l B^l\|_F^2 &= \sum_{i=l+1}^d \sigma_i^2 = \sum_{i=1}^d \sigma_i^2 - \sum_{i=1}^l \sigma_i^2 = \|Y\|_F^2 - \sum_{i=1}^l \sum_{j=1}^n |\langle y_j, u_i \rangle_{\mathbb{R}^m}|^2 \leq \\ &\leq \|Y\|_F^2 - \sum_{i=1}^l \sum_{j=1}^n |\langle y_j, \hat{u}_i \rangle_{\mathbb{R}^m}|^2 = \sum_{i=1}^d \sum_{j=1}^n |C_{ij}^d|^2 - \sum_{i=1}^l \sum_{j=1}^n |C_{ij}^d|^2 = \sum_{i=l+1}^d \sum_{j=1}^n |C_{ij}^d|^2 = \|Y - \hat{U}^l C^l\|_F^2. \end{aligned}$$

□



**Napomena 1.3.1.** (Alternativno traženje POD baze.)  
 Uočimo da vrijedi

$$\begin{aligned}\|Y - \hat{U}^l C^l\|_F^2 &= \sum_{i=1}^m \sum_{j=1}^n |Y_{ij} - \sum_{k=1}^l \hat{U}_{ik}^l C_{kj}|^2 = \sum_{j=1}^n \sum_{i=1}^m |Y_{ij} - \sum_{k=1}^l \langle \hat{u}_k, y_j \rangle_{\mathbb{R}^m} \hat{U}_{ik}^l|^2 \\ &= \sum_{j=1}^n \|y_j - \sum_{k=1}^l \langle y_j, \hat{u}_k \rangle_{\mathbb{R}^m} \hat{u}_k\|_{\mathbb{R}^m}^2.\end{aligned}$$

Analogno

$$\|Y - U^l B^l\|_F^2 = \sum_{j=1}^n \|y_j - \sum_{k=1}^l \langle y_j, u_k \rangle_{\mathbb{R}^m} u_k\|_{\mathbb{R}^m}^2.$$

Korolar 1.3.2 nam daje da vrijedi

$$\sum_{j=1}^n \|y_j - \sum_{k=1}^l \langle y_j, u_k \rangle_{\mathbb{R}^m} u_k\|_{\mathbb{R}^m}^2 \leq \sum_{j=1}^n \|y_j - \sum_{k=1}^l \langle y_j, \hat{u}_k \rangle_{\mathbb{R}^m} \hat{u}_k\|_{\mathbb{R}^m}^2$$

za bilo koji skup  $\{\hat{u}_i\}_{i=1}^l$  ortonormiranih vektora.

Dakle, iz Korolara 1.3.2 slijedi da POD bazu možemo tražiti tako da rješavamo problem

$$\min_{\tilde{u}_1, \dots, \tilde{u}_l \in \mathbb{R}^m} \sum_{j=1}^n \|y_j - \sum_{k=1}^l \langle y_j, \tilde{u}_k \rangle_{\mathbb{R}^m} \tilde{u}_k\|_{\mathbb{R}^m}^2, \quad \text{uz uvjet } \langle \tilde{u}_i, \tilde{u}_j \rangle_{\mathbb{R}^m} = \delta_{ij}, 1 \leq i, j \leq l. \quad (1.7)$$

$\sum_{k=1}^l \langle y_j, \tilde{u}_k \rangle_{\mathbb{R}^m} \tilde{u}_k$  predstavlja ortogonalnu projekciju vektora  $y_j$  na potprostor razapet vektorima  $\tilde{u}_k$ . Ukoliko vektore  $\tilde{u}_k$ ,  $k = 1, \dots, l$  posložimo u matricu  $\tilde{U} = [\tilde{u}_1 \ \tilde{u}_2 \ \dots \ \tilde{u}_l] \in \mathbb{R}^{m \times l}$ , vrijedi  $\sum_{k=1}^l \langle y_j, \tilde{u}_k \rangle_{\mathbb{R}^m} \tilde{u}_k = \tilde{U} \tilde{U}^T y_j$ . Problem (1.7) možemo zapisati kao

$$\min_{\tilde{U} \in \mathbb{R}^{m \times l}} \sum_{j=1}^n \|y_j - \tilde{U} \tilde{U}^T y_j\|_{\mathbb{R}^m}^2 = \min_{\tilde{U} \in \mathbb{R}^{m \times l}} \|Y - \tilde{U} \tilde{U}^T Y\|_F^2, \quad \text{uz uvjet } \tilde{U}^T \tilde{U} = I_l. \quad (1.8)$$

Problem (1.7) je zahvalniji za interpretirati. Naime  $\sum_{k=1}^l \langle y_j, \tilde{u}_k \rangle_{\mathbb{R}^m} \tilde{u}_k$  predstavlja ortogonalnu projekciju vektora  $y_j$  na potprostor razapet s  $\{\tilde{u}_i\}_{i=1}^l$ . Ukoliko od vektora  $y_j$  oduzmemo tu projekciju i računamo kvadrat norme dobivenog vektora, dobivamo udaljenost vektora  $y_j$  od njegove projekcije (na kvadrat). To napravimo za sve vektore  $y_1, \dots, y_n$  i zbrojimo “greške”. Dakle, POD baza najbolje opisuje vektore  $y_1, \dots, y_n$  u smislu najmanjih kvadrata. Ovdje vidimo sličnosti s linearnom regresijom. Ipak, linearna regresija ne “gleda” udaljenosti od ortogonalnih projekcija na potprostor, već vertikalnu udaljenost od regresijskog pravca (ne nužno ortogonalnu projekciju na njega).

Kroz matricu  $\Sigma$  vidimo koliko griješimo pri aproksimaciji podataka  $y_1, \dots, y_n$  projekcijom na potprostor. Iz korolara 1.3.2 proizlazi da minimum problema (1.7) iznosi  $\sum_{i=l+1}^d \sigma_i^2$ . Ukoliko je ta vrijednost malena ne gubimo gotovo ništa aproksimacijom podataka (smanjivanjem dimenzije). U praksi se čak dešava da aproksimacija popravi podatke jer se ovim postupkom uklanja svugdje prisutna greška mjerenja.

**Teorem 1.3.3** (Eckart-Young-Mirsky [5]).

Neka je  $M \in \mathbb{C}^{m \times n}$  kompleksna matrica. Neka je  $M = U\Sigma V^*$  njen SVD,  $\Sigma = \text{diag}(\sigma_i)_{i=1}^{\min(m,n)}$ ,  $\sigma_1 \geq \dots \geq \sigma_{\min(m,n)} \geq 0$ .

Za  $k \in \{1, \dots, \min(m, n)\}$ , definiramo  $U_k = U(:, 1 : k)^{14}$ ,  $\Sigma_k = \Sigma(1 : k, 1 : k)$ ,  $V_k = V(:, 1 : k)$  i  $M_k = U_k \Sigma_k V_k^*$ . Tada je  $M_k$  optimalna aproksimacija ranga  $k$  matrice  $M$  u 2-normi<sup>15</sup> i Frobeniusovoj normi, tj. vrijedi:

$$\min_{\text{rank}(N) \leq k} \|M - N\|_2 = \|M - M_k\|_2 = \sigma_{k+1},$$

$$\min_{\text{rank}(N) \leq k} \|M - N\|_F = \|M - M_k\|_F = \sqrt{\sum_{i=k+1}^{\min(m,n)} \sigma_i^2}.$$

Sada koristeći prethodni teorem i notaciju (1.8) lako dokažemo optimalnost POD baze. Neka je  $Y = U\Sigma V^T$  SVD matrice  $Y$ . Definiramo  $Y_k = U_k \Sigma_k V_k^T$  kao u prethodnom teoremu. Tada vrijedi:

$$\begin{aligned} \|Y - \tilde{U}\tilde{U}^T Y\|_F^2 &\geq (\text{rank}(\tilde{U}\tilde{U}^T Y) \leq k) \geq \|Y - Y_k\|_F^2 = \|Y - U_k \Sigma_k V_k^T\|_F^2 = \|U_k^* Y - \Sigma_k V_k^T\|_F^2 = \\ &= \|Y - U_k U_k^* Y\|_F^2 = \sum_{i=k+1}^d \sigma_i^2. \end{aligned}$$

### Algoritam računanja POD baze ranga $l$

1. Budući da znamo vezu između SVD-a i POD-a, najjednostavniji način je izračunati SVD i uzeti prvih  $l$  stupaca matrice  $U$  za POD bazu. Ipak, to se u praksi ne pokazuje kao najefikasniji način dobivanja POD-baze.
2. Vektori  $\{u_i\}_{i=1}^d$  su svojstveni vektori matrice  $YY^T$ , stoga je drugi način dobivanja POD baze izračun dekompozicije svojstvenih vrijednosti za matricu  $YY^T$  i uzimanje  $l$  svojstvenih vektora uz  $l$  najvećih svojstvenih vrijednosti za POD bazu.

<sup>14</sup>  $U(:, 1 : k)$  predstavlja oznaku za prvih  $k$  stupaca matrice  $U$ ,  $\Sigma(1 : k, 1 : k)$  je oznaka za prvih  $k$  stupaca unutar prvih  $k$  redaka matrice  $\Sigma$ .

<sup>15</sup>  $\|A\|_2 = \max_{\|x\|_{\mathbb{C}^n}=1} \|Ax\|_{\mathbb{C}^m}$ ,  $x \in \mathbb{C}^n$ ,  $A \in \mathbb{C}^{m \times n}$ .

3. U praksi je dimenzija vektora  $y_j$  jako velika i nerijetko vrijedi  $n \ll m$  stoga nam je problem svojstvenih vrijednosti za matricu  $YY^T \in \mathbb{R}^{m \times m}$  skuplje rješavati od problema svojstvenih vrijednosti za matricu  $Y^TY \in \mathbb{R}^{n \times n}$ . Znamo da vrijedi  $Yv_i = \sigma_i u_i$ , stoga rješavamo problem svojstvenih vrijednosti za matricu manje dimenzije. Dobivamo vektore  $v_i$  i svojstvene vrijednosti  $\lambda_i = \sigma_i^2$ . Pomoću gornje jednakosti dobivamo  $u_i = \frac{1}{\sqrt{\lambda_i}} Yv_i$ , za  $\lambda_i > 0$ .

U praksi želimo znati koliko griješimo pri redukciji dimenzije. Da bismo odabrali odgovarajuću dimenziju  $l$  važno je znati slijedeći omjer:  $\varepsilon(l) = \frac{\sum_{i=1}^l \lambda_i}{\sum_{i=1}^d \lambda_i}$ .

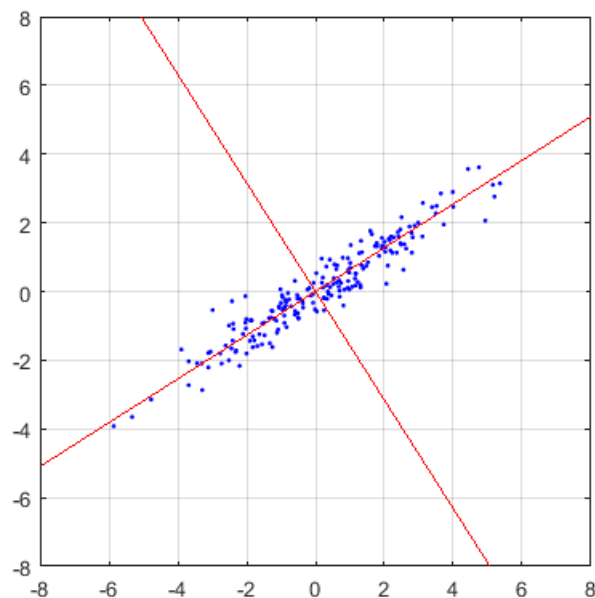
## Primjer u 2D

Uzimamo 200 vektora u  $\mathbb{R}^2$  i crtamo ih u ravnini. Crtamo ih kao točke, a ne kao strelice radi ljepšeg prikaza. Kod je pisan u Matlab-u.

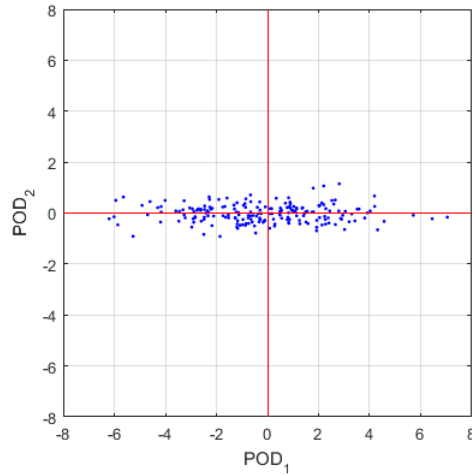
```
L = 200; % broj vektora

% generirajmo vektore
C = [1 2; 1 1];
X = C * randn(2, L);
```

Računamo SVD:  $[U, S, W] = \text{svd}(X)$ ; . Stupci matrice  $U$  su vektori POD baze. Crtamo podatke zajedno s glavnim smjerovima.

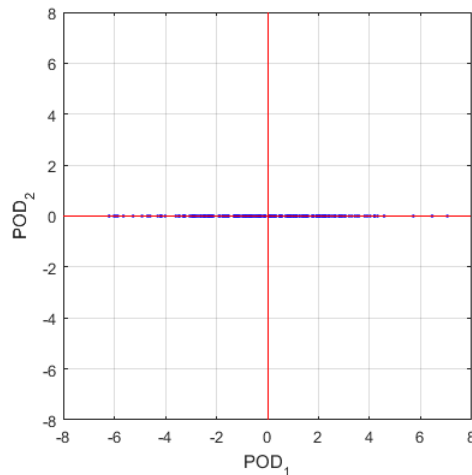


Zapišimo podatke u POD bazi u matricu  $X_U$ :  $X_U = U' * X$ ; . Nacrtajmo rotirane podatke.



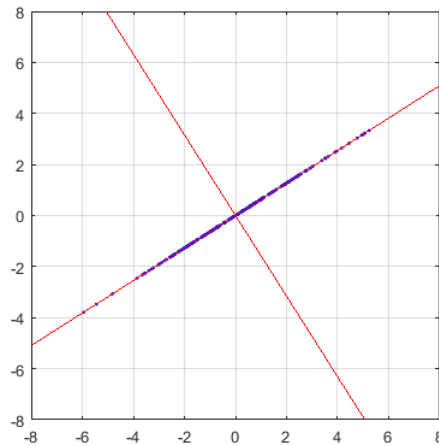
Podaci u POD bazi.

Projicirajmo podatke na prvu glavnu komponentu (1D potprostor):  $X_U(2, :) = 0$ ; . Odmah primjetimo da je suma kvadrata udaljenosti originalnih točaka od njihovih projekcija jednaka  $\sigma_2^2$ . To je posljedica korolara 1.3.2 i napomene iza njega (Metrika se neće promijeniti pri rotaciji).

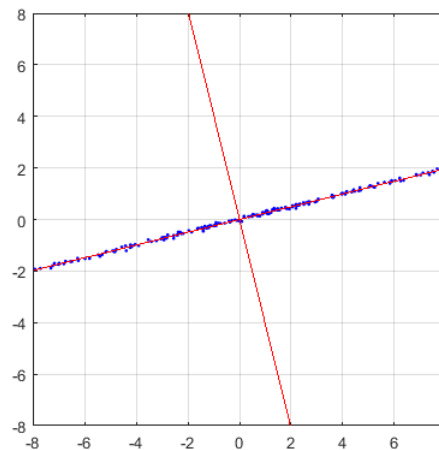


$$\sum_{i=1}^{200} \|x_i - x_{projekcija}\|_{\mathbb{R}^2}^2 = \sigma_2^2 = 25.0458.$$

Vratimo još samo podatke u početnu (kanonsku bazu):  $X_{\text{approx}} = U * X_u ;$ . Usput primijetimo da smo prethodna dva koraka (zapisa u POD bazi i projiciranja) mogli preskočiti i odmah dobiti  $X_{\text{approx}}$  tako da u matricu  $S$  na poziciju (2,2) stavimo  $\sigma_2 = 0$  i pomnožimo  $X_{\text{approx}} = U * S * W'$ . Izvod ove tvrdnje se vidi u dokazu korolara 1.3.2. Naravno, vrijedi i generalizacija na veću dimenziju.



Sada kada vidimo što se u pozadini događa, možemo zaključiti da će  $\sigma_2^2$  (greška) za podatke koji se formiraju oko pravca kroz ishodište uz male perturbacije biti manja nego kod većih perturbacija. Primjer:



$$\sigma_2^2 = 0.4091.$$

## 1.4 DMD (Dynamic Mode Decomposition)

### Ideja i definicija

Slijedeća razrada napravljena je prema [8] i prema [6].

U izgradnji DMD metode pretpostavljamo da su podaci generirani dinamičkim sistemom

$$\frac{d\mathbb{x}(t)}{dt} = \mathbf{f}(\mathbb{x}, t; \mu), \quad \mathbb{x}(t) = [x_1(t), x_2(t), \dots, x_n(t)]^T.$$

$\mathbb{x}(t) \in \mathbb{R}^n$  je vektor koji reprezentira stanje dinamičkog sustava u vremenu  $t$ ,  $\mu$  sadrži parametre sustava i  $\mathbf{f}(\cdot)$  predstavlja dinamiku. Pokrenemo sustav i mjerimo stanje svakih  $\Delta t$  vremenskih jedinica. Dobivamo stanja sustava u diskretnom vremenu i označavamo ih s  $x_k = \mathbb{x}(k\Delta t)$ ,  $k = 0, 1, 2, \dots$ . Diskretnu funkciju evolucije sistema označavamo s  $\mathbf{F}$ :

$$x_{k+1} = \mathbf{F}(x_k).$$

U generalnom slučaju nije moguće konstruirati rješenje za nelinearni sustav pa se koriste numeričke metode za izračun budućih stanja sustava. DMD metoda je sposobna predvidjeti buduće stanje sustava bez da poznaje dinamiku  $\mathbf{f}(\cdot)$ . Sve što nam je potrebno su izmjerena stanja sustava u diskretnim trenucima. DMD metoda lokalno aproksimira nelinearni sustav linearnim sustavom

$$\frac{d\mathbb{x}}{dt} = \mathcal{A}\mathbb{x}, \quad \mathcal{A} \in \mathbb{C}^{n \times n}. \quad (1.9)$$

Rješenje gornjeg sustava znamo i ono je dano s  $\mathbb{x}(t) = e^{\mathcal{A}t}\mathbb{x}(0)$ . Ukoliko podatke uzorkujemo svakih  $\Delta t$  vremenskih jedinica, imamo

$$x_{k+1} = \mathbb{x}((k+1)\Delta t) = e^{\mathcal{A}((k+1)\Delta t)}\mathbb{x}(0) = e^{\mathcal{A}(\Delta t)}e^{\mathcal{A}(k\Delta t)}\mathbb{x}(0) = e^{\mathcal{A}(\Delta t)}\mathbb{x}(k\Delta t) = e^{\mathcal{A}(\Delta t)}x_k.$$

Uz oznaku  $A := e^{\mathcal{A}(\Delta t)} \in \mathbb{C}^{n \times n}$  neprekidni sustav (1.9) možemo opisati analognim diskretnim sistemom uzorkovanim svakih  $\Delta t$  vremenskih jedinica:

$$x_{k+1} = Ax_k.$$

DMD algoritam će nam dati dekompoziciju svojstvenih vrijednosti matrice  $A$  (vodeće svojstvene vektore i svojstvene vrijednosti) koja optimalno pogoduje izmjerenim vrijednostima  $x_k$  ( $k = 0, 1, \dots, m$ ) u smislu najmanjih kvadrata, tj. tako da je vrijednost

$$\sum_{k=0}^{m-1} \|x_{k+1} - Ax_k\|_2^2$$

minimalna.

U cilju minimiziranja gornjeg izraza mjerenja  $x_0, x_1, \dots, x_m$  grupiramo u sljedeće matrice:

$$X = \begin{bmatrix} | & | & & | \\ x_0 & x_1 & \dots & x_{m-1} \\ | & | & & | \end{bmatrix},$$

$$X' = \begin{bmatrix} | & | & & | \\ x_1 & x_2 & \dots & x_m \\ | & | & & | \end{bmatrix}.$$

Lokalno linearnu transformaciju zapisujemo u terminima gornjih matrica na način

$$X' \approx AX.$$

Matrica  $A$  koja minimizira izraz  $\|X' - AX\|_F$  je dana s  $A = X'X^\dagger$  (Napomena 1.2.2 + svojstva transponiranja i pseudoinverza.<sup>16</sup>).

#### Definicija 1.4.1. Dinamička modalna dekompozicija.

Pretpostavimo da imamo dinamički sistem  $\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, t; \mu)$  i dva skupa  $X$  i  $X'$  kao gore tako da vrijedi  $x_{k+1} = \mathbf{F}(x_k)$  gdje je  $\mathbf{F}$  funkcija evolucije sustava u diskretnom vremenu  $\Delta t$ . DMD računa vodeću svojstvenu dekompoziciju linearnog operatora  $A$  koji najbolje opisuje podatke  $X' \approx AX$ :  $A = X'X^\dagger$ . DMD modovi (dinamički modovi) su svojstveni vektori matrice  $A$  i svaki DMD mod odgovara pripadnoj svojstvenoj vrijednosti matrice  $A$ .

Ideja DMD algoritma nije računati matricu  $A$  kao  $X'X^\dagger$  jer to mogu biti matrice jako velikih dimenzija i račun može potrajati poprilično dugo. DMD metoda se vodi činjenicom da rješenje  $\mathbf{x}(t)$  sustava  $\frac{d\mathbf{x}}{dt} = \mathcal{A}\mathbf{x}$  možemo zapisati u bazi svojstvenih vektora matrice  $\mathcal{A}$  i to na način

$$\mathbf{x}(t) = \sum_{k=1}^n \phi_k e^{\omega_k t} b_k = \Phi e^{\Omega t} \mathbf{b},^{17} \quad (1.10)$$

gdje su  $\phi_k$  i  $\omega_k$  svojstveni vektori i svojstvene vrijednosti matrice  $\mathcal{A}$ , a koeficijenti  $b_k$  su koordinate točke  $\mathbf{x}(0)$  u bazi svojstvenih vektora, odnosno  $\mathbf{b} = \Phi^\dagger \mathbf{x}(0)$ .

DMD metoda će dati način da odredimo najznačajnije svojstvene smjerove i aproksimaciju rješenja ćemo tražiti u bazi tih smjerova (vektora). Budući da radimo s diskretnim uzorkom, mi ćemo računati svojstvene vektore matrice  $A := e^{\mathcal{A}(\Delta t)}$ . No, matrica  $\mathcal{A}$  i matrica  $A$

<sup>16</sup> $\|X' - AX\|_F = \|AX - X'\|_F = \|X^T A^T - X'^T\|_F \Rightarrow A^T = (X^T)^\dagger (X')^T = (X^\dagger)^T (X')^T \Rightarrow A = X'X^\dagger$ .

<sup>17</sup> $\Omega = \text{diag}(\omega_1, \dots, \omega_n)$ ,  $\Phi = [\phi_1, \dots, \phi_n]$ ,  $\mathbf{b} = [b_1, \dots, b_n]^T$ .

imaju iste svojstvene vektore jer  $e^{\mathcal{A}(\Delta t)} = e^{\Phi\Omega\Phi^{-1}(\Delta t)} = \Phi e^{\Omega\Delta t}\Phi^{-1}$ . Vidimo da za svojstvene vrijednosti matrice  $A$  vrijedi  $\lambda_k = e^{\omega_k\Delta t}$  pa ćemo iz svojstvenih vrijednosti matrice  $A$  lako dobiti svojstvene vrijednosti matrice  $\mathcal{A}$  kao  $\omega_k = \frac{\ln(\lambda_k)}{\Delta t}$ .

## Izvod algoritma

1. Najprije izračunamo reducirani SVD matrice  $X$  ranga  $r$ , tj. izračunamo matrice  $U \in \mathbb{C}^{n,rx}$ ,  $\Sigma \in \mathbb{C}^{r,rx}$  i  $V \in \mathbb{C}^{m,rx}$  takve da  $X = U\Sigma V^*$ .  $U$  i  $V$  su unitarne matrice. Rang  $r$  određujemo tako da pogledamo singularne vrijednosti nereduciranog SVD-a. Ukoliko se podaci mogu dobro aproksimirati projekcijom na potprostor, kažemo da je prisutna niže dimenzionalna struktura u podacima i tada će singularne vrijednosti u matrici  $\Sigma$  naglo pasti blizu 0 nakon određenog broja dominantnih modova.
2. Računamo projekciju matrice  $A$  na prvih  $r$  POD modova.  
Što bi bila projekcija matrice  $A$ <sup>18</sup>?
  - Rang  $r$  u SVD-u smo odredili tako da je projekcija podataka na POD modove dobra u smislu najmanjih kvadrata, stoga umjesto podataka  $x_k$  gledamo njihove projekcije na potprostor razapet s prvih  $r$  POD modova:  $\tilde{x}_k = U^*x_k$ .
  - Djelovanje sustava ćemo gledati samo na tom potprostoru. Trebamo definirati preslikavanje  $\tilde{A}$  koje dinamiku preslikavanja  $A$  imitira na zadanom potprostoru. Imamo vektor  $\tilde{x}_k$  i želimo ga preslikati u isti potprostor dinamikom  $\tilde{A}$ . Prvo se djelovanjem matrice  $U$  prebacimo u kanonsku bazu prostora  $\mathbb{R}^n$ :  $U\tilde{x}_k$ . Tu djeluje dinamika  $A$  pa njome preslikamo  $U\tilde{x}_k$ :  $AU\tilde{x}_k$ . Zatim se vratimo u bazu potprostora djelovanjem matrice  $U^*$ :  $U^*AU\tilde{x}_k$ . Dakle imamo  $\tilde{x}_{k+1} = U^*AU\tilde{x}_k$  što nam daje  $\tilde{A} = U^*AU$ .
  - Dakle, projekcija matrice  $A$  na prvih  $r$  POD modova je preslikavanje koje imitira preslikavanje  $A$  na potprostoru razapetom s prvih  $r$  modova. To preslikavanje se računa kao  $U^*AU$ . Mi ne znamo matricu  $A$  pa matricu  $\tilde{A}$  računamo kao  $\tilde{A} = U^*AU = U^*X'X^\dagger U = U^*X'V\Sigma^{-1}U^*U = U^*X'V\Sigma^{-1}$ .

Iz  $\tilde{A}v = \lambda v \Rightarrow U^*AUv = \lambda v \Rightarrow A(Uv) = \lambda(Uv)$  vidimo da je svaka svojstvena vrijednost matrice  $\tilde{A}$  ujedno i svojstvena vrijednost matrice  $A$  i da je pripadni svojstveni vektor matrice  $A$  dan s  $Uv$  (što je zapravo samo zapis vektora  $v$  u kanonskoj bazi). Budući da se projiciranjem na POD bazu, ukoliko se može odabrati dobar  $r$  ne gubi puno, nadamo se da ćemo na ovaj način dobiti sve za dinamiku bitne modove.

---

<sup>18</sup>U literaturi poznata kao Rayleighjev kvocijent matrice.



3. Slijedeći korak je odrediti svojstvene vrijednosti i svojstvene vektore matrice  $\tilde{A}$ .  $\tilde{A}W = W\Lambda$ , stupci matrice  $W$  su svojstveni vektori, a  $\Lambda$  je dijagonalna matrica sa pripadnim svojstvenim vrijednostima na dijagonali.
4. U zadnjem koraku rekonstruiramo svojstvene vektore matrice  $A$  kao  $\Phi = UW$ . Ipak, u praksi se DMD modovi (stupci matrice  $\Phi$ ) ne računaju ovako već kao  $\Phi = X'V\Sigma^{-1}W$ . Kad se računaju na ovaj način, obično se nazivaju *egzaktne DMD modovi* dok se kod računanja prvim načinom nazivaju *projicirani DMD modovi*. Iz slijedećeg računa se vidi zašto možemo modove računati kao  $X'V\Sigma^{-1}W$ :

$$\tilde{A} = U^*X'V\Sigma^{-1} \Rightarrow UU^*AU = UU^*X'V\Sigma^{-1} \Rightarrow \quad (1.11)$$

$$\Rightarrow AU = X'V\Sigma^{-1} \Rightarrow AUW = X'V\Sigma^{-1}W.$$

$$\tilde{A}W = W\Lambda \Rightarrow U^*AUW = W\Lambda \Rightarrow AUW = UW\Lambda \quad (1.12)$$

$$\Rightarrow X'V\Sigma^{-1}W \stackrel{(1.11)}{=} A(UW) \stackrel{(1.12)}{=} (UW)\Lambda.$$

Isto tako vidimo da će problem nastati kad je svojstvena vrijednost  $\lambda_k = 0$  jer  $X'V\Sigma^{-1}w_k = 0Uw_k = 0$  pa svojstveni vektor koji pripada svojstvenoj vrijednosti nula računa kao  $\phi_k = Uw_k$ .

Nakon što izračunamo gornju aproksimaciju  $r$  svojstvenih vektora i pripadnih svojstvenih vrijednosti, iz formule (1.10) dobivamo aproksimaciju rješenja sustava sa

$$\mathbb{x}(t) \approx \sum_{k=1}^r \phi_k e^{\omega_k t} b_k.$$

## DMD algoritam u Matlabu

Algoritam zapisujemo u Matlabovu funkciju `DMD.m`. Funkcija ima slijedeće ulaze:

- Matrice  $X$  i  $X'$  u oznakama `x1=X` i `x2=X'`.
- Rank `r` SVD dekompozicije.
- Vremenski raskorak `dt` (gore  $\Delta t$ ) kojim su uzorkovani vektori u matricama  $X$  i  $X'$ .

Funkcija nam vraća slijedeće izlaze:

- Matricu  $\Phi$  koja sadrži DMD modove. Oznaka `Phi=Phi`.
- Vektor  $\omega$  u kojemu je zapisano  $r$  svojstvenih vrijednosti matrice  $\mathcal{A}$ . Oznaka `omega=omega`.

- Vektor  $\lambda$  u kojemu je zapisano  $r$  svojstvenih vrijednosti matrice  $A$ . Oznaka  $\lambda = \lambda$ .
- Vektor  $b$  koji sadrži koordinate točke  $x(0)$  u bazi DMD modova.
- Aproksimacije rješenja sustava u prostoru DMD modova u vremenskim trenucima  $t=0, dt, 2dt, \dots, mdt$ .  $i$ -tu aproksimaciju stavljamo u  $i$ -ti stupac matrice  $X_{dmd}$ .

```
function [Phi, omega, lambda, b, Xdmd] = DMD(X1, X2, r, dt)
    %SVD
    [U, S, V] = svd(X1, 'econ');
    r = min(r, size(U, 2));

    %uzimamo prvih r POD modova
    U_r = U(:, 1:r);
    S_r = S(1:r, 1:r);
    V_r = V(:, 1:r);

    %projekcija matrice A na prvih r POD modova
    Atilde = U_r' * X2 * V_r / S_r;

    %dekompozicija svojstvenih vrijednosti
    [W_r, D] = eig(Atilde);

    %DMD modovi
    Phi = X2 * V_r / S_r * W_r;

    %diskretno vremenske svojstvene vrijednosti
    lambda = diag(D);

    %neprekidno vremenske svojstvene vrijednosti
    omega = log(lambda) / dt;

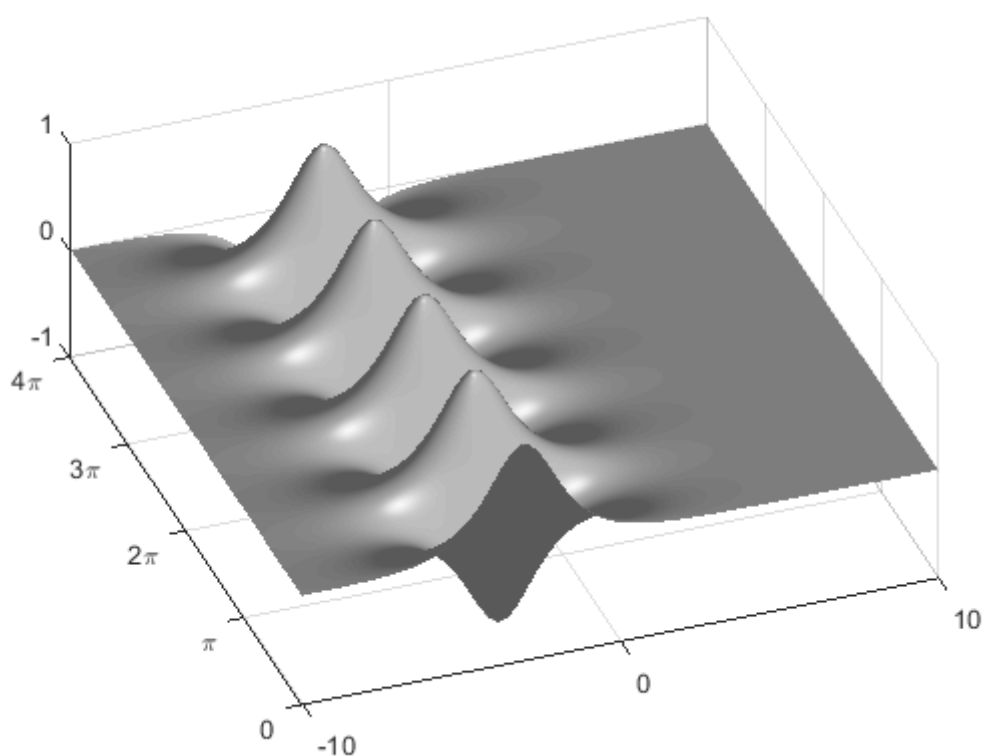
    %koordinate tocke x(0) u bazi Phi
    x1 = X1(:, 1);
    b = Phi \ x1;

    %Aproksimacija vektora u matrici X1 u bazi DMD modova
    mm1 = size(X1, 2); %mm1=m-1
    time_dynamics = zeros(r, mm1);
    t = (0:mm1-1) * dt;
    for iter = 1 : mm1
        time_dynamics(:, iter) = (b. * exp(omega * t(iter)));
    end
    Xdmd = Phi * time_dynamics;
end
```

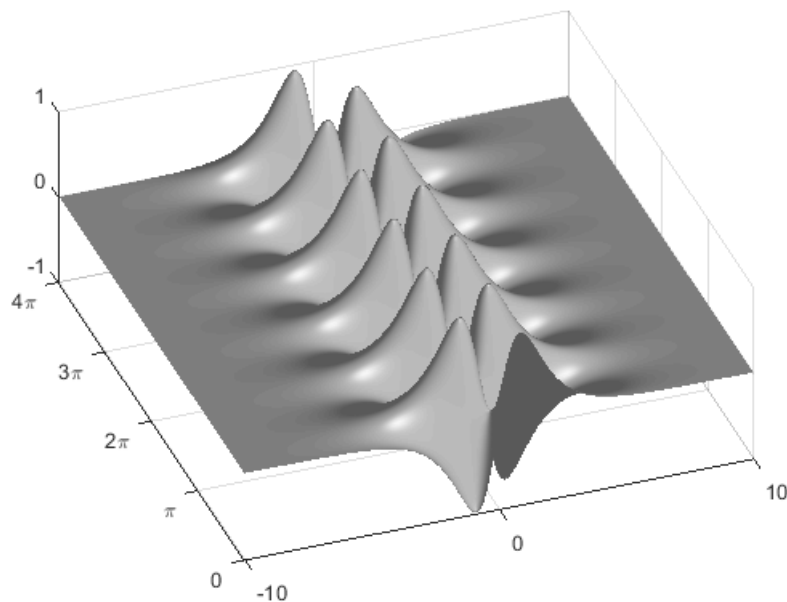
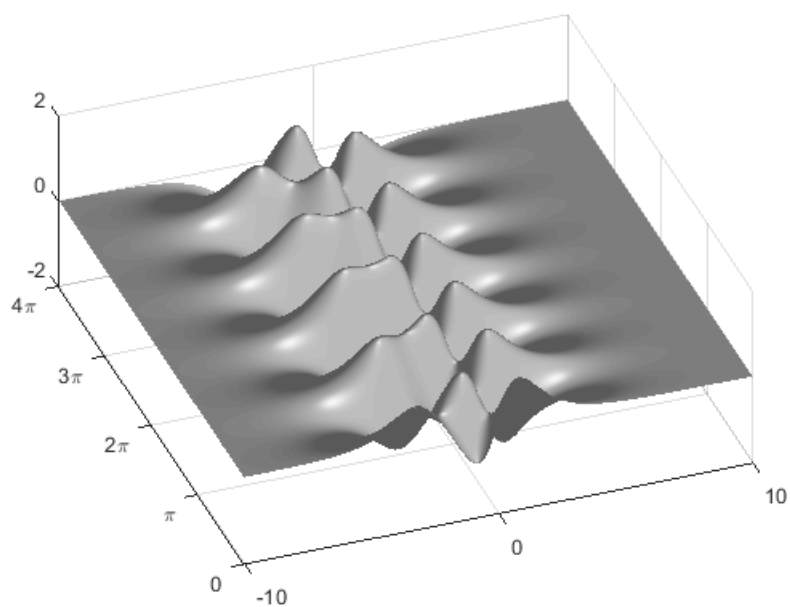
## Primjer DMD dekompozicije

DMD algoritam ćemo demonstrirati na primjeru dva kompleksna prostorno-vremenski isprepletena signala. DMD algoritam ima sposobnost rastaviti signal na sastavne dijelove. Radi usklađenosti s prethodnim poglavljem, možemo pretpostaviti da je signal generiran nelinearnim sustavom  $\frac{dx}{dt} = f(x, t; \mu)$  kojeg ne poznajemo pa ga samim time niti ne znamo riješiti. Ipak, aproksimaciju rješenja možemo dobiti u slijedećih nekoliko trenutaka ukoliko imamo mjerenja svakih  $\Delta t$  trenutaka. DMD metoda će prepoznati dvodimenzionalnu strukturu rješenja, dati modove koji će odgovarati stvarnim komponentama signala i dati aproksimaciju rješenja.

Na plotovima gledamo samo realne dijelove kompleksnih funkcija, no slično vrijedi i za imaginarne dijelove. DMD metoda radi aproksimaciju cijele funkcije, realnog i imaginarnog dijela. Pogledajmo kako izgledaju realni dijelovi podsignala te ukupnog signala:



Slika 1.5:  $f_1(x, t) = \text{sech}(x + 3)e^{i2.3t}$ .

Slika 1.6:  $f_2(x, t) = 2\operatorname{sech}(x)\tanh(x)e^{i2.8t}$ .Slika 1.7:  $f(x, t) = f_1(x, t) + f_2(x, t) = \operatorname{sech}(x + 3)e^{i2.3t} + 2\operatorname{sech}(x)\tanh(x)e^{i2.8t}$ .

Gradimo diskretizaciju prostora  $x$  i vremena  $t$  :

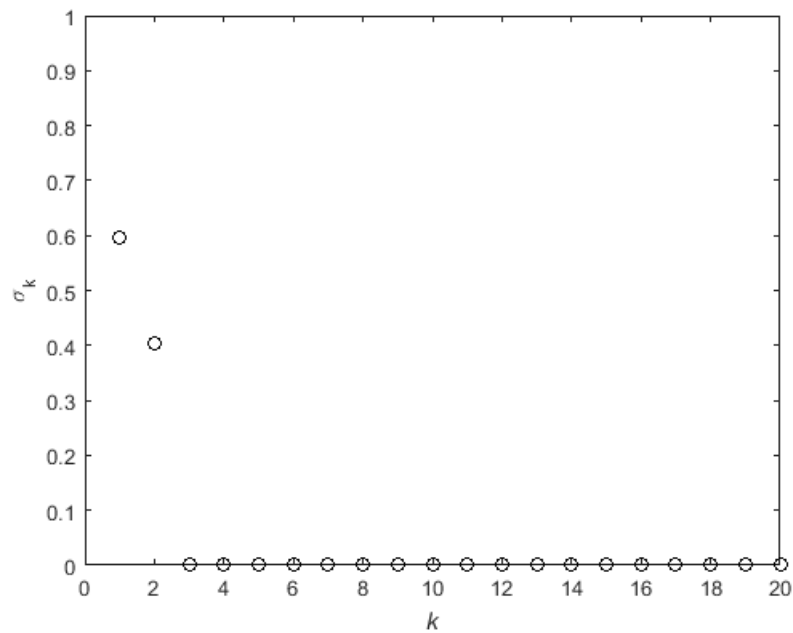
```
x = linspace(-10, 10, 400);
t = linspace(0, 4*pi, 200);
```

Uz gornju diskretizaciju gradimo matrice  $X$  i  $X'$  iz DMD algoritma, oblika

$$X = \begin{bmatrix} | & | & & | \\ f(x, t_1) & f(x, t_2) & \dots & f(x, t_{m-1}) \\ | & | & & | \end{bmatrix},$$

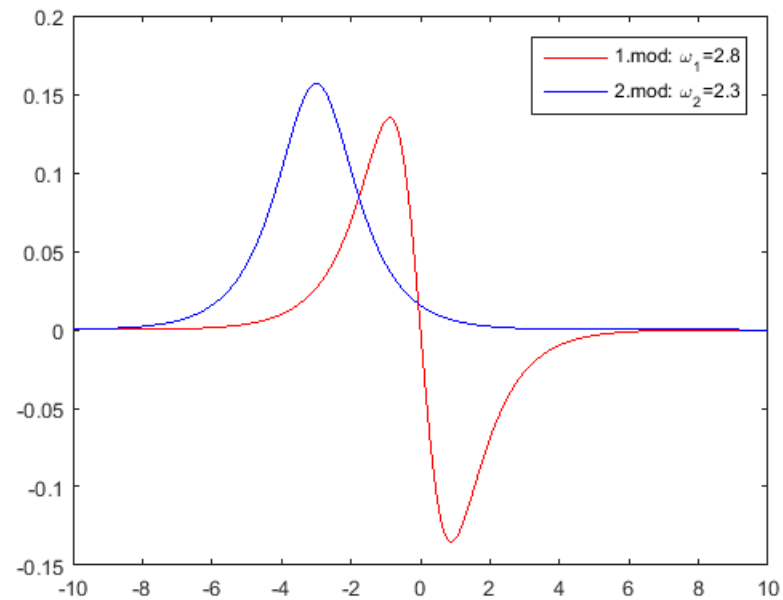
$$X' = \begin{bmatrix} | & | & & | \\ f(x, t_2) & f(x, t_3) & \dots & f(x, t_m) \\ | & | & & | \end{bmatrix}.$$

Prije no provedemo gornji DMD algoritam trebamo odrediti dimenziju uzorka podataka, odnosno  $r$  iz SVD dekompozicije. U tu svrhu plotamo singularne vrijednosti i gledamo kada one padaju na 0.

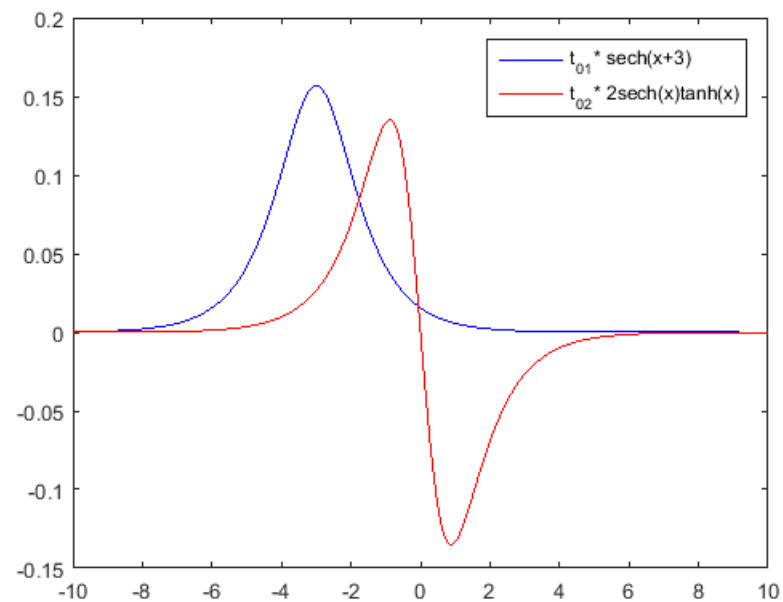


Slika 1.8:  $r=2$ .

Sada imamo sve što nam treba za provođenje opisanog DMD algoritma. Određujemo DMD modove i uspoređujemo ih s originalnim modovima (pojednim signalima).

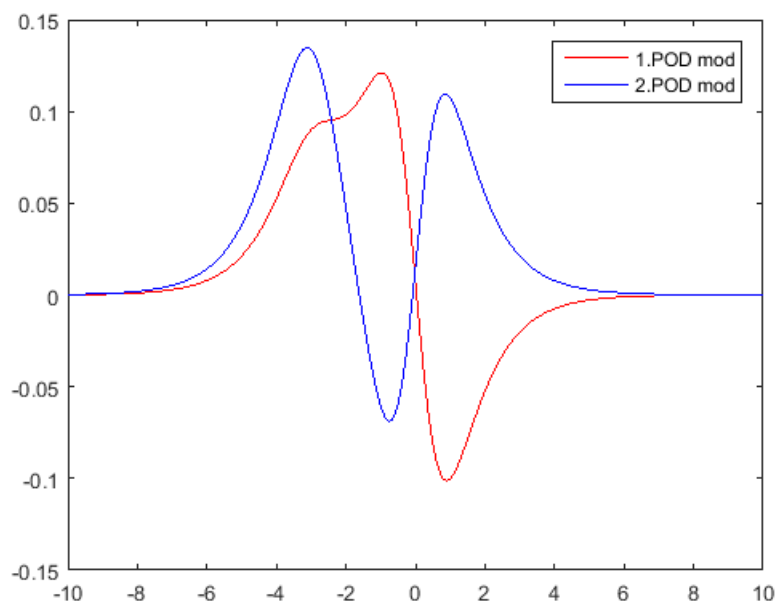


Slika 1.9: DMD modovi i njihove frekvencije.

Slika 1.10: Stvarni modovi (Parametri skaliranja:  $t_{01} = 0.1566$ ,  $t_{02} = -0.135$ .).

Modovi (svojstveni vektori) su intuitivno funkcije koje se kreću sinusoidalno kroz vrijeme. To su upravo funkcije  $f_1(x, 0)$  i  $f_2(x, 0)$  (možemo gledati skalirane funkcije s bilo kojim parametrom  $t \in \langle -1, 0 \rangle \cup \langle 0, 1 \rangle$  jer se zapravo radi o svojstvenim vektorima).  $\omega_1$  i  $\omega_2$  predstavljaju frekvenciju (svojstvene vrijednosti  $\omega_1 = 2.8i$  i  $\omega_2 = 2.3i$ ) kojom se pojedini signal (mod) kreće kroz vrijeme. DMD nam u ovom primjeru uspješno vrati egzaktne modove i frekvencije.

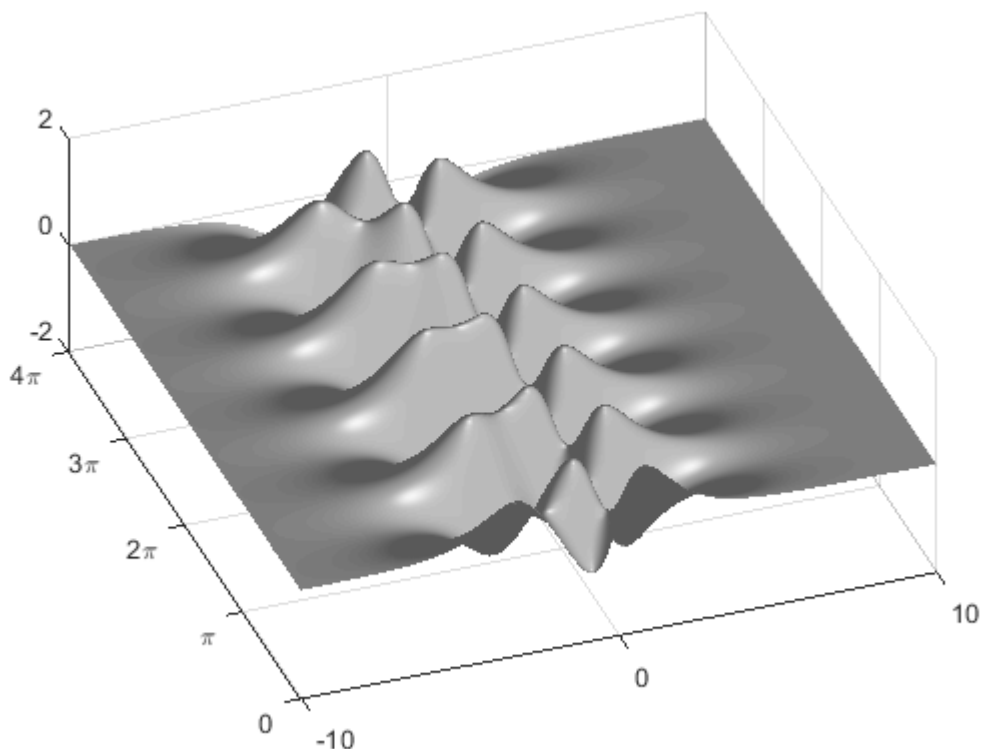
Za usporedbu pogledajmo POD modove:



Slika 1.11: POD modovi.

Primjećujemo da se POD modovi uvelike razlikuju od egzaktnih i DMD modova. To smo mogli i očekivati jer POD modovi gledaju samo prostornu komponentu signala <sup>19</sup>. DMD promatra i prostornu i vremensku komponentu i hvata svu dinamiku ovog sustava. Za kraj pogledajmo kako izgleda aproksimacija rješenja i primijetimo da je pogreška pri aproksimaciji DMD metodom minimalna.

<sup>19</sup>POD modovi su stupci matrice  $U$  iz SVD dekompozicije.



Slika 1.12:  $\|X_{egz} - X_{dmd}\|_F = 6.2026 * 10^{-12}$ .

U ovom primjeru nam je rješenje( $f(x, t)$ ) generirano nepoznatim sustavom poznato, no u stvarnim primjenama to rješenje ne znamo. Imamo samo mjerenja sustava u diskretnim trenucima. DMD metoda će nam dati frekvencije i modove koje ćemo moći iskoristiti za lokalnu aproksimaciju rješenja. Dodatno ćemo ovom metodom i reducirati dimenziju sustava.

Iz ovog primjera bi se dalo naslutiti da bi mogli dobiti i više od lokalnog rješenja, no to nije slučaj. Ovaj je primjer savršeno naštiman za DMD i služi samo da vidimo kako metoda radi. Ipak, metoda se pokazuje jako korisnom u primjenama i vrlo dobro lokalno aproksimira rješenje sustava čija rješenja (a niti nužno jednadžbe) ne znamo.



## Poglavlje 2

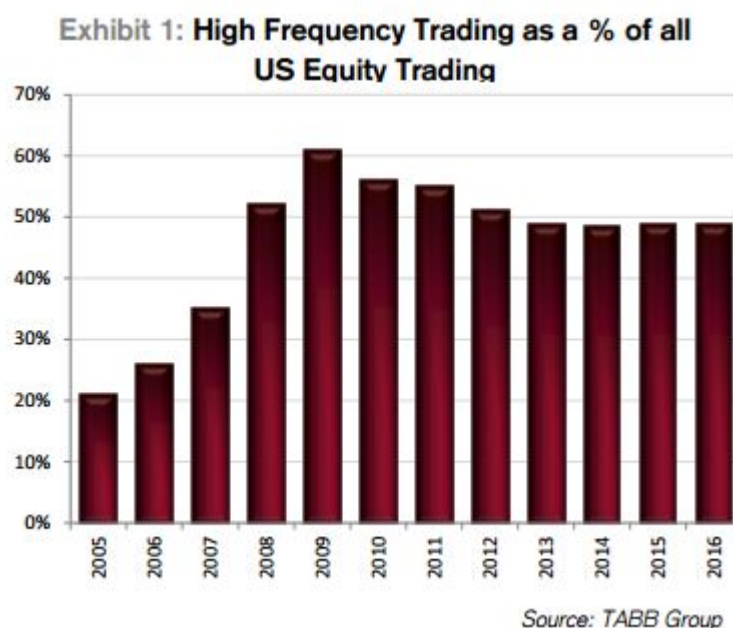
# Primjena DMD-a u algoritamskom trgovanju

Ideje u ovom poglavlju su preuzete iz [7] i [6].

Ukoliko promotrimo DMD algoritam, vidimo da je za njegovo pokretanje dovoljno imati mjerenja dinamičkog sustava u jednako razmaknutim vremenskim trenucima. Dobar primjer takvih mjerenja su dnevne cijene dionica. DMD algoritmom ćemo pokušati uhvatiti dinamiku kretanja dnevnih cijena dionica i predvidjeti kolika će biti cijena slijedeći dan. Budući da uspješnim predviđanjem cijene možemo dobro zaraditi, pogledat ćemo kakva bi nam zarada uistinu bila da smo ulagali u dionice za koje DMD metoda predviđa najveći prinos.

### 2.1 Algoritamsko trgovanje - općenito

Algoritamski trgovati znači izgraditi model koji će predvidjeti cijenu dionice u slijedećem trenutku te na osnovu tog predviđanja prepustiti računalu da automatski donese odluku o kupnji/prodaji financijskog instrumenta. Kad gledamo dnevne podatke, možemo i sami iskoristiti predviđanje modela te na osnovu toga nazvati brokera i obaviti kupnju ili prodaju, no to nije moguće ukoliko radimo s visoko frekventnim podacima (npr. svake sekunde ili milisekunde). Za obavljanje kupnje ili prodaje svake sekunde potrebno je da kompjuterski program komunicira sa automatskim brokerom (Automated Broker Interface). Razvojem automatskog brokera omogućeno je da se cijeli postupak upravljanja portfeljom automatizira i da se iskoriste prednosti visokofrekventnog trgovanja. Pogledajmo koliki je volumen visokofrekventnog trgovanja u svim transakcijama na dioničkom tržištu u SAD-u:



Slika 2.1: <sup>1</sup>Volumen visokofrekventog trgovanja je veći kada je tržište volatilnije.

Da bismo se bavili visokofrekventim trgovanjem potrebna je infrastruktura koja omogućava in-live trgovanje, također je potreban pristup visokofrekventim podacima koji uglavnom nisu besplatni. Automatski broker naplaćuje svaku transakciju koju obavlja. Sve te troškove treba uzeti u obzir pri donošenju odluke o bavljenju ovakvom vrstom trgovanja.

## 2.2 Priprema podataka za DMD

Dnevne podatke o cijenama dionica preuzimamo iz [1]. Nakon što odaberemo koje ćemo dionice promatrati, konstruiramo matricu  $X$ :

$$X = \begin{bmatrix} & \vdots & \\ \dots & x_{ij} & \dots \\ & \vdots & \end{bmatrix},$$

gdje  $x_{ij}$  predstavlja cijenu dionice  $i$  na dan  $j$ . Dakle,  $j$  ti stupac sadrži cijene skupa dionica na dan  $j$ , a  $i$ -ti redak predstavlja cijene dionice  $i$  za promatrano razdoblje. Takva matrica je u obliku kakav nam treba za pokrenuti DMD algoritam.

<sup>1</sup>Slika preuzeta s <https://www.businessinsider.com>.

## 2.3 Trening algoritma za trgovanje

Već smo rekli da ćemo za predikciju cijena dionica koristiti DMD algoritam, no nećemo ga koristiti na bilo koji način. Imamo slobodu izabrati koliko ćemo trenutaka gledati u prošlost da bismo odredili cijenu u budućnosti, također možemo odabrati i koliko daleko gledamo u budućnost. Zbog toga našu DMD metodu parametriziramo sa slijedeća dva cijelobrojna parametra:

$m$  = broj dana u prošlosti koje koristimo za predikciju,

$l$  = broj dana u budućnosti za koje predviđamo cijene dionica.

Trening algoritma je ljepši naziv za određivanje najbolje kombinacije  $(m, l)$  za koju će DMD metoda najbolje predvidjeti cijene dionica u slijedećih nekoliko dana. Uzimamo  $m$  i  $l$  iz nekih dovoljno velikih raspona i pokrenemo DMD( $m, l$ ) za sve kombinacije  $(m, l)$ . Na dostupnim povijesnim podacima vidimo za koje  $(m, l)$  metoda najbolje predviđa rast ili pad cijena dionica. Logično je pretpostaviti da će razni gospodarski sektori imati različitu dinamiku. Zbog toga je za svaki gospodarski sektor potrebno odrediti parametre  $(m, l)$ . Pogledajmo, primjerice kako bi izgledao trening algoritma za sektor izgradnje. Promatramo 8 dionica iz sektora izgradnje:

- D. R. Horton, Inc.



([https://en.wikipedia.org/wiki/D.\\_R.\\_Horton](https://en.wikipedia.org/wiki/D._R._Horton))<sup>2</sup>,

- Lennar



(<https://www.brandsoftheworld.com/logo/lennar>),

- PulteGroup, Inc.



(<https://millbridge-nc.com/pulte-homes/>),

- Toll Brothers, Inc.



(<https://www.builderonline.com/firms/toll-brothers>),

- NVR, Inc.



(<https://www.builderonline.com/firms/nvr>),

- The Home Depot, Inc.



(<https://commons.wikimedia.org/wiki/File:TheHomeDepot.svg>),

<sup>2</sup>U zgrade navodimo odakle su preuzeta loga tvrtki.

- Lowe's Companies, Inc.



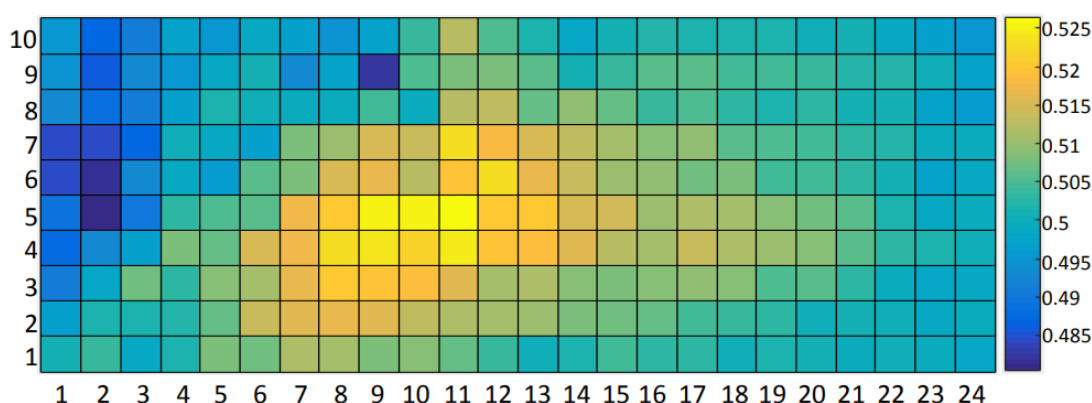
(<https://www.lowes.com/l/logos.html>),

- The Sherwin-Williams Company



(<https://www.brandsoftheworld.com/logo/sherwin-williams-9>).

Uzimamo  $m \in \{1, 2, \dots, 25\}$  i  $l \in \{1, 2, \dots, 10\}$ . Gledamo kako algoritam predviđa kretanje cijena dionica u određenom vremenskom periodu i crtamo graf:



Slika 2.2: <sup>3</sup>Heat plot.

Svaki kvadratić na gornjem plotu predstavlja postotak kojim metoda pogađa dali će cijene dionica rasti ili padati. Primjerice, najžutiji kvadratić na polju (11, 5) znači da DMD algoritam koji gleda cijene prošlih 11 dana i predviđa cijene slijedećih 5 dana točno predvidi smjer kretanja cijena dionica u 52,5% slučajeva što znači da će, ukoliko iskoristimo predviđanja algoritma, 52,5% naših ulaganja donijeti zaradu.

Posebno zanimljivi su nam sektori kod kojih se primjećuju veća povezana područja s postotkom pogotka iznad 50% kao što vidimo žuto područje na gornjem grafu (engleski naziv za takva područja je *hot – spots*).

## 2.4 Mogući algoritmi za trgovanje

U prethodnom poglavlju vidimo kako možemo odrediti parametre  $(m, l)$  naše  $DMD(m, l)$  metode. Ta saznanja možemo koristiti na više načina. Primjerice:

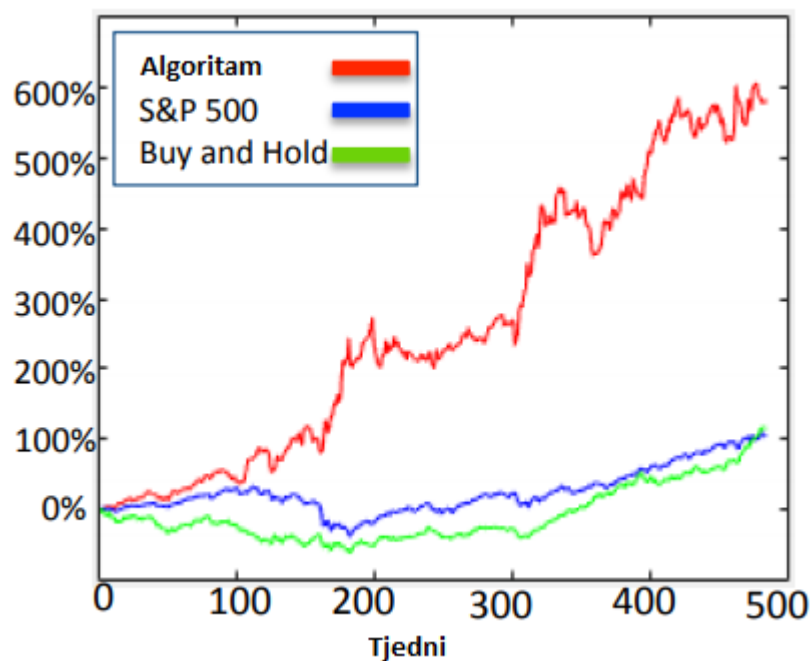
<sup>3</sup>Slika preuzeta iz [7].

- Možemo gledati kako su se cijene kretale prethodnih 10 godina (ako su nam ti podaci dostupni) i za to razdoblje određujemo parametre naše metode. Ovdje je važno napraviti kros validaciju, odnosno pogledati dali se hot-spotovi pomiču znatnije kada gledamo različite vremenske raspone. U tu svrhu uzimamo vremenski okvir od 2 godine i za podatke u tom okviru gledamo gdje se nalaze hot-spotovi. Kako imamo podatke za prethodnih 10 godina, možemo na više načina uzeti podatke raspona dvije godine i pogledati hot-spotove. Ukoliko primjetimo da su hot-spotovi robustni na promjene podataka istog raspona (2 godine), možemo pretpostaviti da će u slijedećem razdoblju oni također ostati robustni te ćemo imati parametre za našu metodu. To je naravno par  $(m, l)$  za koji je postotak pogotka najveći (i veći od 50%). U ovom algoritmu jednom odredimo parametre metode i koristimo ih za sva slijedeća predviđanja.
- U realnoj situaciji hot-spotovi se mijenjaju, a ponekad nemamo dovoljno podataka da bi napravili kros validaciju i odredili robustne parametre metode. Naš algoritam sada može funkcionirati na način da gledamo prethodnih 100 dana da bi odredili parametre metode danas. Na taj način parametre metode stalno prilagođavamo, odnosno nanovo ih određujemo u odnosu na to koji parametri su imali najbolje predviđanje kretanja cijena prethodnih 100 dana. Te parametre koristimo za predviđanje cijena za slijedećih nekoliko dana te krećemo iznova.
- Autori u radu [7] navode da hot spotovi mogu iščeznuti i da naša predviđanja kretanja cijena u tom slučaju neće biti dobra. U takvom slučaju prethodni algoritam neće biti dobar. Moramo ga prilagoditi da trgujemo samo onda kada se pojave hot-spotovi. Zbog toga definiramo hot-spot kao par cijelih brojeva  $(m, l)$  za koji vrijedi:

- $S_{(m,l)} > t$ ,
- $\frac{1}{9} \sum_{j=-1}^1 \sum_{k=-1}^1 S_{(m+k, l+j)} > t$ ,

gdje  $S_{(m,l)}$  predstavlja postotak pogotka DMD( $m, l$ ) metode, a  $t$  predstavlja proizvoljno određeni postotak pogotka iznad kojeg želimo trgovati uzimajući u obzir predviđanja metode (primjerice 53%). Sada ćemo naša predviđanja kretanja cijena dionica koristiti za trgovanje samo onda kada se pojavi hot spot u matrici pogotka (odnosno gornjem heat plotu). Na ovaj način možemo potpuno automatizirati naše trgovanje. Nema potrebe da gledamo heat plot sada kada znamo odrediti hot spot automatski.

Testiranjem unatrag pogledajmo kakvi bi povrati bili da smo koristili gore opisani algoritam za trgovanje gore navedenim dionicama iz građevinskog sektora u periodu 10 godina. Za usporedbu vidimo kakvi bi povrati bili da smo ulagali u index S&P500 ili da smo kupili i držali dionice u koje ulažemo.



Slika 2.3: <sup>4</sup>Anualizirani povrat 21.48%.

Iz gornjeg grafa vidimo da se korištenjem DMD metode mogu ostvariti ozbiljni povrati. Dakle, DMD metoda može stati uz bok pa čak i nadmašiti statističke i vjerojatnosne modele koji su uobičajeni pri odabiru strategije ulaganja. Napomenimo još da je u ovom radu korišten najosnovniji oblik DMD metode, no razvijene su i nadogradnje same metode (npr. DMD s kontrolom) koje se također mogu iskoristiti za algoritamsko trgovanje i potencijalno ostvariti još veće povrate.

---

<sup>4</sup>Slika preuzeta iz [7].

## Poglavlje 3

# Proširena dinamička modalna dekompozicija

U ovom poglavlju uvodimo pojam Koopmanovog operatora i opisujemo EDMD (Extended Dynamic Mode Decomposition) metodu pomoću koje računamo svojstvene vrijednosti, svojstvene funkcije i modove Koopmanovog operatora. Pokazujemo vezu između DMD algoritma i EDMD algoritma te zaključujemo da je DMD metoda samo specijalan slučaj EDMD metode. Vodimo se idejama opisanim u radu [10].

### 3.1 Koopmanov operator

Neka je  $\mathcal{M} \subseteq \mathbb{R}^N$  prostor stanja,  $n \in \mathbb{Z}$  i  $\mathbf{F} : \mathcal{M} \rightarrow \mathcal{M}$  operator evolucije sistema. Operator  $\mathbf{F}$  preslikava stanje  $\mathbf{x}(n) \in \mathcal{M}$  u stanje  $\mathbf{x}(n+1) \in \mathcal{M}$ . Za razliku od operatora  $\mathbf{F}$ , Koopmanov operator  $\mathcal{K}$  djeluje na prostoru  $\mathcal{F}$  skalarnih funkcija definiranih na prostoru stanja  $\mathcal{M}$ . Točnije, Koopmanov operator preslikava funkciju  $\phi \in \mathcal{F}$ ,  $\phi : \mathcal{M} \rightarrow \mathbb{C}$  na način

$$\mathcal{K}\phi = \phi \circ \mathbf{F}, \quad (3.1)$$

gdje  $\circ$  predstavlja kompoziciju funkcija. Dakle, Koopmanov operator preslikava funkciju prostora stanja u funkciju prostora stanja.

Funkcije prostora stanja  $\phi : \mathcal{M} \rightarrow \mathbb{C}$  nazivamo *opservacijama*. Koopmanov operator definira novi dinamički sustav  $(\mathcal{F}, n, \mathcal{K})$  koji opisuje dinamiku kretanja opservacija u diskretnom vremenu. Operator  $\mathcal{K}$  djeluje na funkcijama te je stoga beskonačnodimenzijsan čak i kad je  $\mathbf{F}$  konačnodimenzijsan. Operator  $\mathcal{K}$  je također linearan čak i kad je  $\mathbf{F}$  nelinearan. Beskonačnodimenzijsnost Koopmanovog operatora potencijalno može biti problematična, no ukoliko ga možemo aproksimirati konačnodimenzijsnim operatorom uz malu grešku dobivamo konačnodimenzijsan linearan problem koji znamo riješiti.

Dinamički sustav definiran operatorom  $\mathbf{F}$  i dinamički sustav definiram Koopmanovim operatorom  $\mathcal{K}$  predstavljaju dvije različite parametrizacije iste dinamike. Vezu između dvije parametrizacije predstavlja vektorska funkcija  $\mathbf{g}(\mathbb{x}) = \mathbb{x}$  koju nazivamo *opservacija čitavog prostora* i  $\{(\mu_k, \varphi_k, \mathbf{v}_k)\}_{k=1}^K$  skup od  $K$  trojki Koopmanovih svojstvenih vrijednosti, svojstvenih funkcija i modova potrebnih za rekonstrukciju funkcije  $\mathbf{g}$ .  $K$  je najčešće beskonačan.

Rekli smo da  $\mathcal{K}$  preslikava skalarnu funkciju u skalarnu funkciju.  $\mathbf{g}$  je vektorska funkcija, no njene komponente su skalarne funkcije  $\mathbf{g} = (g_1, g_2, \dots, g_N)^\top$ ,  $g_i \in \mathcal{F}$ . Komponente  $g_i$  možemo prikazati u prostoru svojstvenih funkcija Koopmanovog operatora na način  $g_i = \sum_{k=1}^K v_{ik} \varphi_k$ ,  $v_{ik} \in \mathbb{C}$ . Slažemo vektore  $\mathbf{v}_j = [v_{1j}, v_{2j}, \dots, v_{Nj}]^\top$  te dobivamo

$$\begin{aligned} \mathbb{x} = \mathbf{g}(\mathbb{x}) &= \begin{bmatrix} g_1(\mathbb{x}) \\ g_2(\mathbb{x}) \\ \vdots \\ g_N(\mathbb{x}) \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^K v_{1k} \varphi_k(\mathbb{x}) \\ \sum_{k=1}^K v_{2k} \varphi_k(\mathbb{x}) \\ \vdots \\ \sum_{k=1}^K v_{Nk} \varphi_k(\mathbb{x}) \end{bmatrix} = \\ &= \varphi_1(\mathbb{x}) \begin{bmatrix} v_{11} \\ v_{21} \\ \vdots \\ v_{N1} \end{bmatrix} + \varphi_2(\mathbb{x}) \begin{bmatrix} v_{12} \\ v_{22} \\ \vdots \\ v_{N2} \end{bmatrix} + \dots + \varphi_K(\mathbb{x}) \begin{bmatrix} v_{1K} \\ v_{2K} \\ \vdots \\ v_{NK} \end{bmatrix} = \sum_{k=1}^K v_k \varphi_k(\mathbb{x}), \end{aligned}$$

vektor  $v_k$  nazivamo  $k$ -ti Koopmanov mod. U gornjem razvoju smo pretpostavili da se skalarne funkcije  $g_i$  nalaze u potprostoru razapetom s  $K$  svojstvenih funkcija, no nismo pretpostavili da svojstvene funkcije čine bazu za prostor  $\mathcal{F}$  već samo da razapinju neki njegov potprostor.

Sada buduće stanje sustava možemo promatrati direktno kroz evoluciju stanja  $\mathbb{x}$  ili kroz evoluciju opservacije  $\mathbf{g}$  čitavog prostora:

$$\mathbf{F}(\mathbb{x}) = (\mathcal{K}\mathbf{g})(\mathbb{x}) = \sum_{k=1}^K v_k (\mathcal{K}\varphi_k)(\mathbb{x}) = \sum_{k=1}^K \mu_k v_k \varphi_k(\mathbb{x}).$$

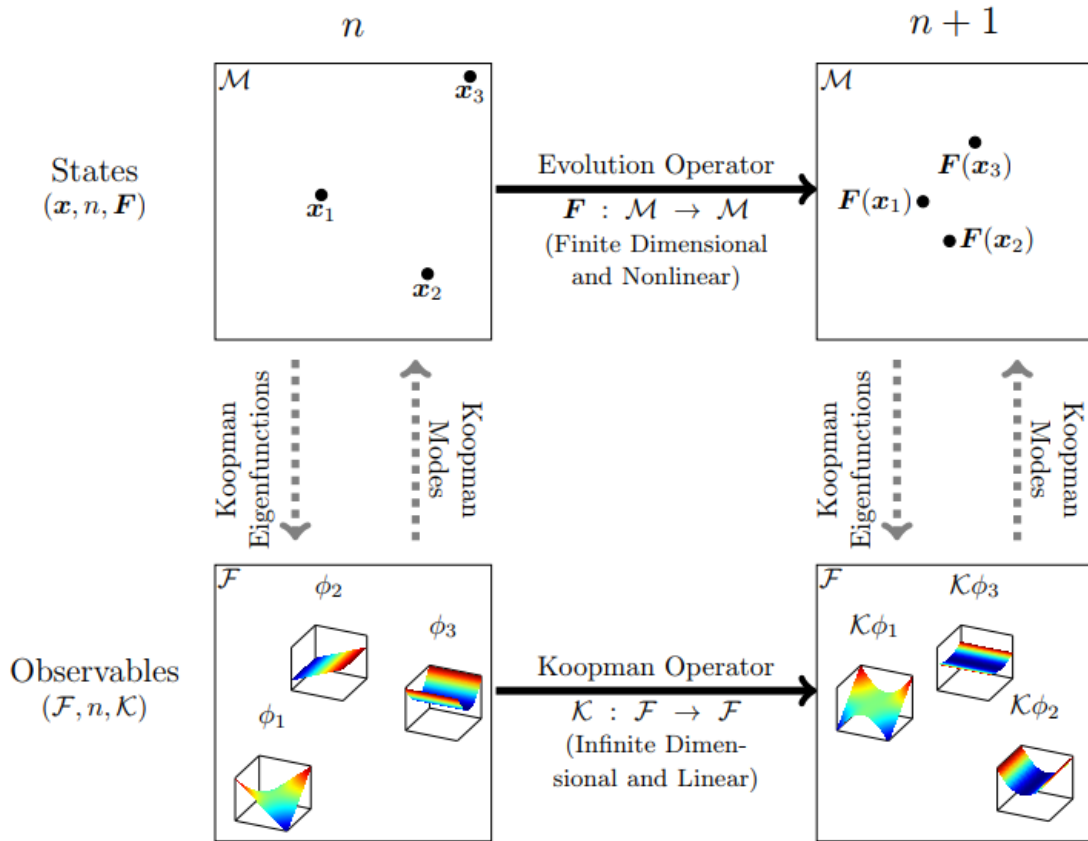
Gornja reprezentacija dinamike  $\mathbf{F}(\mathbb{x})$  ima prednost jer je dinamika vezana uz svojstvenu funkciju određena pripadnom svojstvenom vrijednošću. Sve zajedno možemo pogledati na slijedećem diagramu. Ukoliko želimo iz stanja  $\mathbb{x}$  izračunati  $(\mathcal{K}\phi)(\mathbb{x})$ , možemo to postići na dva načina:

- $\phi \rightarrow \mathcal{K}\phi \rightarrow \mathcal{K}\phi(\mathbb{x})$
- $\mathbb{x} \rightarrow \mathbf{F}(\mathbb{x}) \rightarrow \phi(\mathbf{F}(\mathbb{x})) = \mathcal{K}\phi(\mathbb{x})$



Također, stanje  $\mathbf{F}(\mathbf{x})$  možemo izračunati na dva načina:

- $\mathbf{x} \rightarrow \mathbf{F}(\mathbf{x})$
- $\mathbf{x} = \mathbf{g}(\mathbf{x}) \rightarrow (\mathcal{K}\mathbf{g})(\mathbf{x}) = \mathbf{F}(\mathbf{x})$



Slika 3.1: <sup>1</sup>Djelovanje Koopmanovog operatora.

Vidimo da možemo birati želimo li raditi s konačnodimenzionalnim nelinearnim sustavom ili s beskonačnodimenzionalnim linearnim sustavom ovisno o tome kojim nam se putem na gornjem dijagramu lakše kretati. Kada funkciju  $\mathbf{g}$  (opservacija čitavog prostora) zapišemo u terminima Koopmanovih svojstvenih funkcija, zamjenjujemo kompleksnu (nelinearnu) dinamiku stanja  $\mathbf{x}$  s linearnom dinamikom funkcija  $\{\varphi_i\}_i$ .

<sup>1</sup>Slika je preuzeta iz [10].

### 3.2 EDMD (Extended Dynamic Mode Decomposition)

EDMD je metoda kojom aproksimiramo Koopmanov operator. Time također dobivamo svojstvene vrijednosti, svojstvene funkcije i modove Koopmanovog operatora. EDMD metoda zahtijeva:

- kao i DMD metoda, skup parova  $\{(\mathbb{x}_i, \mathbb{y}_i) : \mathbb{x}_i, \mathbb{y}_i \in \mathcal{M}, \mathbb{y}_i = \mathbf{F}(\mathbb{x}_i)\}_{i=1}^M$  od kojih formiramo matrice  $X = [\mathbb{x}_1, \mathbb{x}_2, \dots, \mathbb{x}_M]$  i  $Y = [\mathbb{y}_1, \mathbb{y}_2, \dots, \mathbb{y}_M]$  (Napomenimo samo da ovdje podatci nisu uzorkovani svakih  $\Delta t$  vremenskih jedinica kao kod DMD metode, već su uzorkovani iz neke distribucije na  $\mathcal{M}$ )
- riječnik opservacija  $\mathcal{D} = \{\psi_1, \psi_2, \dots, \psi_K\}$ ,  $\psi_i \in \mathcal{F}$  koje razapinju potprostor  $\mathcal{F}_{\mathcal{D}} \subset \mathcal{F}$ .

Također definiramo vektorsku funkciju  $\Psi : \mathcal{M} \rightarrow \mathbb{C}^{1 \times K}$  kao

$$\Psi(\mathbb{x}) = [\psi_1(\mathbb{x}), \psi_2(\mathbb{x}), \dots, \psi_K(\mathbb{x})].$$

Želimo generirati  $\mathbf{K} \in \mathbb{R}^{K \times K}$ , konačno dimenzionalnu aproksimaciju<sup>2</sup> Koopmanova operatora  $\mathcal{K}$ . Funkciju  $\phi \in \mathcal{F}_{\mathcal{D}}$  možemo zapisati kao

$$\phi = \sum_{k=1}^K a_k \psi_k = \Psi \mathbf{a}.$$

Budući da  $\mathcal{F}_{\mathcal{D}}$  nije nužno invarijantan potprostor operatora  $\mathcal{K}$ ,  $\mathcal{K}\phi$  možemo zapisati kao

$$\mathcal{K}\phi = (\Psi \circ \mathbf{F})\mathbf{a} = \Psi(\mathbf{K}\mathbf{a}) + r, \quad (3.2)$$

gdje je  $\mathbf{K}$   $K \times K$  matrica, a  $r \in \mathcal{F}$  rezidual.

Matrica  $\mathbf{K}$  nam je nepoznata i nju želimo odrediti. To ćemo napraviti tako da minimiziramo sumu kvadrata reziduala:

$$\begin{aligned} J &= \frac{1}{2} \sum_{m=1}^M |r(\mathbb{x}_m)|^2 = \frac{1}{2} \sum_{m=1}^M |((\Psi \circ \mathbf{F})(\mathbb{x}_m) - \Psi(\mathbb{x}_m)\mathbf{K})\mathbf{a}|^2 = \\ &= \frac{1}{2} \sum_{m=1}^M |(\Psi(\mathbb{y}_m) - \Psi(\mathbb{x}_m)\mathbf{K})\mathbf{a}|^2. \end{aligned}$$

---

<sup>2</sup> $\mathbf{K}$  predstavlja matrični zapis operatora  $P_{\mathcal{F}_{\mathcal{D}}} \mathcal{K}|_{\mathcal{F}_{\mathcal{D}}} : \mathcal{F}_{\mathcal{D}} \rightarrow \mathcal{F}_{\mathcal{D}}$ , gdje je  $\mathcal{K}|_{\mathcal{F}_{\mathcal{D}}}$  restrikcija Koopmanovog operatora na  $\mathcal{F}_{\mathcal{D}}$  i  $P_{\mathcal{F}_{\mathcal{D}}}$  ortogonalna projekcija na  $\mathcal{F}_{\mathcal{D}}$ .

Gornji problem najmanjih kvadrata ima jedinstveno rješenje. Minimum se postiže u

$$\mathbf{K} = \mathbf{G}^\dagger \mathbf{A},$$

gdje  $\dagger$  predstavlja generalizirani inverz, a matrice  $\mathbf{G}$  i  $\mathbf{A}$  su definirane kao

$$\mathbf{G} = \frac{1}{M} \sum_{m=1}^M \Psi(\mathbf{x}_m)^* \Psi(\mathbf{x}_m), \quad \mathbf{G} \in \mathbb{C}^{K \times K},$$

$$\mathbf{A} = \frac{1}{M} \sum_{m=1}^M \Psi(\mathbf{x}_m)^* \Psi(\mathbf{y}_m), \quad \mathbf{A} \in \mathbb{C}^{K \times K}.$$

Kao rezultat dobivamo konačnodimenzionalnu aproksimaciju,  $\mathbf{K}$ , Koopmanova operatora  $\mathcal{K}$ .  $\mathbf{K}$  preslikava funkciju  $\phi \in \mathcal{F}_{\mathcal{D}}$  u funkciju  $\hat{\phi} \in \mathcal{F}_{\mathcal{D}}$ .

Ako je  $\xi_j$   $j$ -ti svojstveni vektor od  $\mathbf{K}$  sa svojstvenom vrijednošću  $\mu_j$ , onda je EDMD aproksimacija svojstvene funkcije Koopmanova operatora dana s

$$\varphi_j = \Psi \xi_j,$$

što vidimo uvrštavanjem u (3.2) ( $\mathcal{K}\varphi_j = \mathcal{K}\Psi\xi_j \approx \Psi(\mathbf{K}\xi_j) = \Psi(\mu_j\xi_j) = \mu_j\Psi\xi_j = \mu_j\varphi_j$ ).

### Aproksimacija Koopmanovih modova

Iz formule

$$\mathbf{x} = \mathbf{g}(\mathbf{x}) = \sum_{k=1}^K v_k \varphi_k(\mathbf{x}), \quad \mathbf{g} : \mathcal{M} \rightarrow \mathbb{R}^N$$

vidimo da su Koopmanovi modovi (kako smo ih gore definirali) vektori  $v_k$  koji su nam potrebni za rekonstrukciju opservacije čitavog prostora u bazi svojstvenih funkcija Koopmanovog operatora. Opservaciju čitavog prostora možemo prikazati kao

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} g_1(\mathbf{x}) \\ g_2(\mathbf{x}) \\ \vdots \\ g_N(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \mathbf{e}_1^* \mathbf{x} \\ \mathbf{e}_2^* \mathbf{x} \\ \vdots \\ \mathbf{e}_N^* \mathbf{x} \end{bmatrix},$$

gdje je  $\mathbf{e}_i$   $i$ -ti jedinični vektor u  $\mathbb{R}^N$ . Sada pretpostavljamo da su sve  $\mathbf{g}_i \in \mathcal{F}_{\mathcal{D}}$ , tako da  $\mathbf{g}_i = \sum_{k=1}^K \psi_k b_{k,i} = \Psi \mathbf{b}_i$  (ukoliko to nije slučaj, Koopmanove modove ćemo aproksimirati

tako da funkcije  $\mathbf{g}_i$  projiciramo na  $\mathcal{F}_{\mathcal{D}}$  te će kvaliteta aproksimacije ovisiti o izboru riječnika  $\mathcal{D}$ ). Sad opservaciju čitavog prostora možemo zapisati kao

$$\mathbf{g} = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_N \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^K \psi_k b_{k,1} \\ \sum_{k=1}^K \psi_k b_{k,2} \\ \vdots \\ \sum_{k=1}^K \psi_k b_{k,N} \end{bmatrix} = \begin{bmatrix} \Psi \mathbf{b}_1 \\ \Psi \mathbf{b}_2 \\ \vdots \\ \Psi \mathbf{b}_N \end{bmatrix} = \mathbf{B}^T \Psi^T = (\Psi \mathbf{B})^T, \quad \mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_N] \in \mathbb{C}^{K \times N}.$$

Zapišimo sada funkcije  $\psi_i$  u terminima numerički aproksimiranih svojstvenih funkcija Koopmanovog operatora. Definirajmo najprije vektorsku funkciju  $\Phi : \mathcal{M} \rightarrow \mathbb{C}^{1 \times K}$  na način

$$\Phi(\mathbb{x}) = [\varphi_1(\mathbb{x}), \varphi_2(\mathbb{x}), \dots, \varphi_K(\mathbb{x})].$$

Iz prethodnog poglavlja znamo da je aproksimacija svojstvenih funkcija Koopmanovog operatora dana s  $\varphi_i = \Psi \xi_i$ , gdje je  $\xi_i \in \mathbb{C}^K$  i-ti svojstveni vektor operatora  $\mathbf{K}$  pridružen svojstvenoj vrijednosti  $\mu_i$ . Matrično to možemo zapisati kao  $\Phi = \Psi \Xi$ , gdje je  $\Xi = [\xi_1, \xi_2, \dots, \xi_K]$ . Da bismo  $\Psi$  zapisali u terminima  $\Phi$  potrebno je invertirati matricu  $\Xi$

$$\Xi^{-1} = \mathbf{W}^* = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_K]^*,$$

gdje je  $\mathbf{w}_i$  lijevi svojstveni vektor<sup>3</sup> matrice  $\mathbf{K}$  pridružen svojstvenoj vrijednosti  $\mu_i$  (tj.  $\mathbf{w}_i^* \mathbf{K} = \mathbf{w}_i^* \mu_i$ ) skaliran tako da vrijedi  $\mathbf{w}_i^* \xi_i = 1$ . Slijedi,

$$\begin{aligned} \mathbf{g} &= \mathbf{B}^T \Psi^T, \quad \Psi = \Phi \Xi^{-1} = \Phi \mathbf{W}^* \implies \mathbf{g} = \mathbf{B}^T (\Phi \mathbf{W}^*)^T = (\mathbf{W}^* \mathbf{B})^T \Phi^T \\ \implies \mathbf{g} &= \mathbf{V} \Phi^T = \sum_{k=1}^K \mathbf{v}_k \varphi_k, \quad \mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_K] = (\mathbf{W}^* \mathbf{B})^T, \end{aligned}$$

gdje je  $\mathbf{v}_i = (\mathbf{w}_i^* \mathbf{B})^T$  i-ti Koopmanov mod.

Rezimirajmo,

EDMD metoda zahtijeva skup parova  $\{(\mathbb{x}_i, \mathbb{y}_i) : \mathbb{x}_i, \mathbb{y}_i \in \mathcal{M}, \quad \mathbb{y}_i = \mathbf{F}(\mathbb{x}_i)\}_{i=1}^M$  koje slažemo u matrice  $X$  i  $Y$ . EDMD metoda također zahtijeva riječnik opservacija  $\mathcal{D}$ . U metodi pretpostavljamo da su vodeće svojstvene funkcije Koopmanovog operatora sadržane (uz malu grešku) u  $\mathcal{F}_{\mathcal{D}}$ , potprostoru razapetom elementima iz  $\mathcal{D}$ . Uzimajući u obzir ove informacije možemo izračunati konačnodimenzionalnu aproksimaciju Koopmanova operatora,  $\mathbf{K} = \mathbf{G}^\dagger \mathbf{A}$ . Svojstvene vrijednosti operatora  $\mathbf{K}$  nam daju aproksimaciju svojstvenih vrijednosti Koopmanovog operatora. Desni svojstveni vektori operatora  $\mathbf{K}$  generiraju aproksimaciju svojstvenih funkcija Koopmanovog operatora dok lijevi svojstveni vektori od  $\mathbf{K}$  generiraju aproksimaciju Koopmanovih modova.

<sup>3</sup> $\xi_i$  je sv. vektor matrice  $\mathbf{K}$  sa sv. vrijednošću  $\mu_i$ ,  $i = 1, \dots, K \implies \mathbf{K} \Xi = \Xi M$ ,  $M = \text{diag}(\mu_1, \dots, \mu_K)$   
 $\Xi^{-1} / \mathbf{K} \Xi = \Xi M / \Xi^{-1} \implies \Xi^{-1} \mathbf{K} = M \Xi^{-1} \implies \mathbf{W}^* \mathbf{K} = M \mathbf{W}^* \implies \mathbf{w}_i^* \mathbf{K} = \mathbf{w}_i^* \mu_i$ ,  $i = 1, \dots, K$ .

### 3.3 Veza između EDMD-a i DMD-a

U ovom poglavlju ćemo pokazati da je DMD metoda samo specijalan slučaj EDMD metode. Ukoliko uzmemo specijalan riječnik  $\mathcal{D}$  za EDMD metodu, EDMD metoda i DMD metoda će nam dati iste svojstvene vrijednosti i iste modove za dani skup parova  $\{(\mathbb{x}_i, \mathbb{y}_i) : \mathbb{x}_i, \mathbb{y}_i \in \mathcal{M}, \mathbb{y}_i = \mathbf{F}(\mathbb{x}_i)\}_{i=1}^M$  odnosno za dane matrice  $X$  i  $Y$ .

DMD modove smo definirali kao svojstvene vektore matrice  $\mathbf{K}_{DMD} = YX^\dagger$ , gdje je  $j$ -tom modu pridružena  $j$ -ta svojstvena vrijednost  $\mu_j$ .

**Teorem 3.3.1.** *Neka je  $\mathcal{D} = \{\mathbf{e}_1^*, \mathbf{e}_2^*, \dots, \mathbf{e}_N^*\}$  naš riječnik, gdje je  $\mathbf{e}_i^* \mathbb{x} = x_i$ ,  $\mathbb{x} \in \mathbb{R}^N$ . Koopmanovi modovi dobiveni EDMD metodom uz ovakav riječnik su ujedno i svojstveni vektori matrice  $\mathbf{K}_{DMD}$ , odnosno oni su ujedno i DMD modovi.*

*Dokaz.* Opservaciju čitavog prostora smo definirali kao

$$\mathbf{g}(\mathbb{x}) = \begin{bmatrix} g_1(\mathbb{x}) \\ g_2(\mathbb{x}) \\ \vdots \\ g_N(\mathbb{x}) \end{bmatrix} = \begin{bmatrix} \mathbf{e}_1^* \mathbb{x} \\ \mathbf{e}_2^* \mathbb{x} \\ \vdots \\ \mathbf{e}_N^* \mathbb{x} \end{bmatrix}.$$

Trebamo matricu težina  $\mathbf{B} \in \mathbb{C}^{N \times N}$  takvu da vrijedi  $\mathbf{g} = \mathbf{B}^T \Psi^T$ . Slijedi da je  $\mathbf{B}$  jedinična matrica.

$$\implies \mathbf{B} = \mathbf{I} \in \mathbb{C}^{N \times N}.$$

Matricu  $\mathbf{G}$  smo definirali kao

$$\mathbf{G} = \frac{1}{M} \sum_{m=1}^M \Psi(\mathbb{x}_m)^* \Psi(\mathbb{x}_m).$$

Uzmimo u obzir riječnik  $\mathcal{D}$ , slijedi:

$$\begin{aligned} \mathbf{G} &= \frac{1}{M} \sum_{m=1}^M [x_{1m}, x_{2m}, \dots, x_{Nm}]^* [x_{1m}, x_{2m}, \dots, x_{Nm}] = \frac{1}{M} \sum_{m=1}^M \begin{bmatrix} x_{1m}^* x_{1m} & \dots & x_{1m}^* x_{Nm} \\ & \ddots & \\ x_{Nm}^* x_{1m} & \dots & x_{Nm}^* x_{Nm} \end{bmatrix} \\ \implies \mathbf{G}_{ij} &= \frac{1}{M} (x_{i1}^* x_{j1} + x_{i2}^* x_{j2} + \dots + x_{im}^* x_{jm}) \implies \mathbf{G}_{ij}^T = \frac{1}{M} (x_{i1} x_{j1}^* + x_{i2} x_{j2}^* + \dots + x_{im} x_{jm}^*) \\ \implies \mathbf{G}^T &= \frac{1}{M} X X^*. \end{aligned}$$

Analogno dobijemo  $\mathbf{A}^T = \frac{1}{M} YX^*$ . Također, iz izvoda EDMD metode znamo da su Koopmanovi modovi dani kao  $\mathbf{v}_i = (\mathbf{w}_i^* \mathbf{B})^T$ , gdje je  $\mathbf{w}_i$  lijevi svojstveni vektor matrice  $\mathbf{K}$ .

$$\mathbf{B} = \mathbf{I} \implies \mathbf{v}_i^T = \mathbf{w}_i^*.$$

Dobivamo:

$$\mathbf{K}^T = (\mathbf{G}^\dagger \mathbf{A})^T = \mathbf{A}^T \mathbf{G}^{T\dagger} = \frac{1}{M} YX^* \left( \frac{1}{M} XX^* \right)^\dagger = \frac{1}{M} YX^* \mathcal{M}(XX^*)^\dagger = YX^\dagger = \mathbf{K}_{DMD}.$$

Sada vidimo da vrijedi

$$\mathbf{K}_{DMD} \mathbf{v}_i = (\mathbf{v}_i^T \mathbf{K}_{DMD}^T)^T = (\mathbf{w}_i^* \mathbf{K})^T = (\mu_i \mathbf{w}_i^*)^T = \mu_i \mathbf{v}_i.$$

□

Kao što i sam naziv govori, o EDMD-u možemo razmišljati kao o proširenju DMD metode. Kvaliteta aproksimacije Koopmanovih modova ovisi o izboru riječnika  $\mathcal{D}$ . Izborom prikladnijeg riječnika možemo dobiti bolju aproksimaciju u odnosu na aproksimaciju dobivenu DMD-om. Uz pravi izbor riječnika, EDMD metodu možemo primijeniti na širi spektar problema u odnosu na DMD metodu koja implicitno pretpostavlja prilično jednostavan riječnik. Kao dobar izbor riječnika za probleme definirane na  $\mathbb{R}^N$  pokazuju se Hermitovi polinomi.

## Zaključak

U ovom radu smo pokazali da se DMD može uspješno upotrijebiti pri razvoju metoda za algoritamsko trgovanje. Algoritam je sposoban uhvatiti dinamiku kretanja cijena dionica i predvidjeti cijenu u slijedećih nekoliko trenutaka što možemo iskoristiti za zaradu. Napomenimo još jednom da je opisana metoda samo osnova za mnoge nadogradnje koje su razvijene i za one koje su još uvijek u razvoju. Svaka nadogradnja DMD-a ima potencijala za postizanje dobrih rezultata na tržištu dionica ili nekom drugom financijskom tržištu. Jedna od nadogradnji DMD metode je i EDMD metoda koja je opisana u ovom radu. Da bismo EDMD primijenili na cijene dionica, ukoliko želimo postići bolji rezultat od DMD metode, potrebno je uložiti trud pri odabiru pogodnog riječnika za financijsko tržište. Također, DMD s kontrolom je sposoban prepoznati vanjske utjecaje na financijsko tržište što rezultira još boljim modelom za predviđanje cijena. DMD metoda i njene nadogradnje imaju svijetlu budućnost kako u financijama, tako i u drugim područjima primjene.

# Bibliografija

- [1] *Yahoo Finance*, <https://finance.yahoo.com/>.
- [2] Adi Ben-Israel i Thomas NE Greville, *Generalized inverses: theory and applications*, sv. 15, Springer Science & Business Media, 2003.
- [3] Stephen L Campbell i Carl D Meyer, *Generalized inverses of linear transformations*, sv. 56, SIAM, 2009.
- [4] Zlatko Drmac, Igor Mezic i Ryan Mohr, *Data driven modal decompositions: analysis and enhancements*, SIAM Journal on Scientific Computing **40** (2018), br. 4, A2253–A2285.
- [5] Carl Eckart i Gale Young, *The approximation of one matrix by another of lower rank*, Psychometrika **1** (1936), br. 3, 211–218.
- [6] J Nathan Kutz, Steven L Brunton, Bingni W Brunton i Joshua L Proctor, *Dynamic mode decomposition: data-driven modeling of complex systems*, sv. 149, SIAM, 2016.
- [7] Jordan Mann i J Nathan Kutz, *Dynamic mode decomposition for financial trading strategies*, Quantitative Finance **16** (2016), br. 11, 1643–1655.
- [8] Peter J Schmid, *Dynamic mode decomposition of numerical and experimental data*, Journal of fluid mechanics **656** (2010), 5–28.
- [9] Stefan Volkwein, *Proper orthogonal decomposition: Theory and reduced-order modelling*, Lecture Notes, University of Konstanz **4** (2013), br. 4.
- [10] Matthew O Williams, Ioannis G Kevrekidis i Clarence W Rowley, *A data-driven approximation of the koopman operator: Extending dynamic mode decomposition*, Journal of Nonlinear Science **25** (2015), br. 6, 1307–1346.

# Sažetak

U ovom radu opisujemo DMD metodu koju koristimo za aproksimaciju rješenja nelinearnog dinamičkog sustava. Analiziramo teorijsku pozadinu, intuitivnu interpretaciju i algoritam za računanje metode. Također opisujemo POD metodu koja se nalazi u pozadini DMD algoritma. U kratkim crtama ističemo važnost SVD dekompozicije matrice koja nam daje POD modove i generalizirani inverz matrice.

POD metoda nalazi, u smislu najmanjih kvadrata, najbolji potprostor za dani skup vektora. Takav potprostor je pogodan za aproksimaciju sustava većeg ranga sustavom nižeg ranga. DMD metoda pronalazi koherentne prostorno-vremenske modove koji tvore gibanje. Oni su dani svojstvenim vektorima matrice  $A$  koja opisuje lokalno lineariziranu dinamiku sustava. Operator preslikavanja  $A$  općenito je nepoznat, no poznata su izmjerena stanja sustava koja su nam dovoljna za rekonstrukciju dinamike. Iz izmjerenih stanja možemo odrediti operator  $\tilde{A}$  koji je aproksimacija nižeg reda operatora  $A$ . Svojstvene vrijednosti operatora  $\tilde{A}$  čine podskup svojstvenih vrijednosti operatora  $A$ , a iz svojstvenih vektora operatora  $\tilde{A}$  lako rekonstruiramo pripadne svojstvene vektore operatora  $A$  koje nazivamo DMD modovima. Aproksimaciju stanja sustava dobivamo iz DMD modova, pripadnih svojstvenih vrijednosti i početnog stanja.

DMD metodu primjenjujemo na cijene dionica. Hvatamo dinamiku kretanja cijena i predviđamo kuda će cijene ići slijedećih nekoliko trenutaka. Koristeći ta predviđanja razvijamo strategiju ulaganja i u konačnici dobivamo uspješnu metodu algoritamskog trgovanja.

U posljednjem poglavlju DMD metodu stavljamo u širi teoretski kontekst. Uvodimo pojam Koopmanovog operatora i opisujemo EDMD metodu pomoću koje računamo svojstvene vrijednosti, svojstvene funkcije i modove Koopmanovog operatora. Pokazujemo vezu između DMD algoritma i EDMD algoritma te zaključujemo da je DMD metoda samo specijalan slučaj EDMD metode.



# Summary

In this paper we have studied DMD method used to approximate nonlinear dynamics system solution. We analyze the theoretical background, intuitive interpretation and a numerical algorithm. We also describe the POD method which appears in the background of the DMD algorithm. We emphasize the importance of the SVD matrix decomposition which gives us POD modes and the generalized inverse matrix.

POD method finds, in terms of least squares, the best subspace for a given set of vectors. Such a subspace is suitable for the approximation of a higher ranking system with a lower ranking system.

The DMD method finds coherent spatio-temporal modes that form the motion. They are given by the eigenvectors of the matrix  $A$  that describes the locally linearized dynamics of the system. The mapping operator  $A$  is generally unknown, but we know the measured system states that are sufficient for the reconstruction of the dynamics. We can determine the operator  $\tilde{A}$  (lower rank approximation of  $A$ ) using only measured states. The eigenvalues of the operator  $\tilde{A}$  are a subset of the eigenvalues of the operator  $A$ , and from the eigenvectors of operator  $\tilde{A}$  we easily reconstruct the corresponding eigenvectors of operator  $A$ . That eigenvectors are called DMD modes. The proximal state of the system is obtained from DMD modes, associated eigenvalues and initial states.

We apply the DMD method to stock prices. We capture the dynamics of price movements and predict where the price will go in the next few moments. Using that forecasting, we develop an investment strategy and ultimately get a successful method of algorithmic trading.

In the last chapter we put the DMD method into a broader theoretical context. We introduce the Koopman operator and describe the EDMD method by which we calculate eigenvalues, eigenfunctions and modes of Koopman operator. We show the link between the DMD algorithm and the EDMD algorithm and we conclude that DMD method is only a special case of EDMD method.

# Životopis

Ivan Kulušić rođen je 26. lipnja 1993. godine u Šibeniku, Hrvatska. U Šibeniku je pohađao Osnovnu školu "Fausta Vrančića" te "Ekonomsku školu Šibenik". Školovanje nastavlja u Zagrebu na Prirodoslovno-matematičkom fakultetu gdje 2016. godine završava Preddiplomski sveučilišni studij Matematika te upisuje Diplomski studij Financijska i poslovna matematika.